



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV AUTOMOBILNÍHO A DOPRAVNÍHO INŽENÝRSTVÍ

INSTITUTE OF AUTOMOTIVE ENGINEERING

REKONSTRUKCE POVRCHU VOZOVKY

RECONSTRUCTION OF THE ROAD SURFACE

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Andrej Šulák

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Lubor Zháňal, Ph.D.

BRNO 2019

Zadání diplomové práce

Ústav: Ústav automobilního a dopravního inženýrství
Student: **Bc. Andrej Šulák**
Studijní program: Strojní inženýrství
Studijní obor: Automobilní a dopravní inženýrství
Vedoucí práce: **Ing. Lubor Zháňal, Ph.D.**
Akademický rok: 2018/19

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Rekonstrukce povrchu vozovky

Stručná charakteristika problematiky úkolu:

Prostřednictvím měření jízdních výšek a polohy vozidla je možné určit okamžité kontaktní body mezi koly vozidla a vozovkou. Opakovaným průjezdem vybrané trati vzniká mračno bodů, které je možno vhodným zpracováním využít pro rekonstrukci původního povrchu.

Cíle diplomové práce:

Zpracování záznamu jízdy vozidla a vytvoření mračna kontaktních bodů s vozovkou.
Vyhodnocení kontaktních bodů a vytvoření algoritmu pro rekonstrukci povrchu vozovky.
Vytvoření polygonální sítě ve formátu STL.
Analýza dosažené přesnosti.
Návrh technik pro modelaci nezmapovaných vedlejších povrchů za účelem vizualizace.

Seznam doporučené literatury:

REIMPELL, Jornsens. The Automotive Chassis. 2nd edition. Oxford: Butterworth - Heinemann, 2001. 444 s. ISBN 0-7506-5054-0.

GILLESPIE, Thomas. D. Fundamentals of Vehicle Dynamics. Warrendale: Society of Automotive Engineers, 1992. 519 s. ISBN 1-56091-199-9.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2018/19

V Brně, dne

L. S.

prof. Ing. Josef Štětina, Ph.D.
ředitel ústavu

doc. Ing. Jaroslav Katolický, Ph.D.
děkan fakulty

ABSTRAKT

Táto práca pojednáva o rozličných prístupoch k problematike rekonštrukcie povrchu trate na základe rôznych na to určených algoritmov. Taktiež okrem metód rekonštrukcie trate predstavuje aj návrhy na doplnenie polygonálnej siete do nezosnímaných oblastí okolo trate.

KLÚČOVÉ SLOVÁ

mračno bodov, polygonálna sieť, rekonštrukcia, odšumovanie, trať, cesta, doplnenie siete

ABSTRACT

This thesis presents various approaches for race track surface reconstruction based on different algorithms designated for these purposes. In addition to surface reconstruction it also offers proposals for polygon mesh filling in the unscanned areas around the track.

KEYWORDS

point cloud, polygon mesh, reconstruction, noise reduction, track, road, mesh filling

BIBLIOGRAFICKÁ CITÁCIA

ŠULÁK, A. *Rekonstrukce povrchu vozovky*. Brno, 2019. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav automobilního a dopravního inženýrství. 62 s. Vedoucí diplomové práce Lubor Zháňal.

ČESTNÉ PREHLÁSENIE

Prehlasujem, že táto práca je mojím pôvodným dielom, spracoval som ju samostatne pod vedením Ing. Lubora Zháňala, Ph.D. a s použitím literatúry uvedenej v zozname.

V Brne dňa 24. mája 2019

.....

Andrej Šulák

POĎAKOVANIE

V prvom rade by som sa chcel poďakovať môjmu vedúcemu Ing. Luborovi Zháňalovi, Ph.D. za jeho ochotu a odborné rady pri spracovávaní tejto práce. Taktiež veľká vďaka patrí mojim rodičom, ktorí mi vždy pridávali motiváciu ísť vpred a rovnako prejavovali veľkú podporu.

OBSAH

Úvod	9
1 Základy skenovania objektu	10
1.1 Základná terminológia	10
1.2 Metódy skenovania	15
1.3 Metodika spracovania dát	18
2 Metóda získania dát	27
3 Rekonštrukcia trate	29
3.1 Analýza vstupných dát a určenie potrebných metód	29
3.2 Prístup 1 – Rekonštrukcia pomocou bilaterálneho filtra	32
3.3 Prístup 2 – Rekonštrukcia pomocou Laplacoveho odšumovania	41
3.4 Prístup 3 – Rekonštrukcia pomocou prirodzeného súradného systému	44
3.5 Porovnanie a vyhodnotenie jednotlivých metód rekonštrukcie	48
4 Návrh doplnenia okolitého terénu	51
4.1 Vizualizačné doplnenie	51
4.2 Doplnenie pomocou výškových máp	53
5 Použitie nástroja	55
Záver	57
Zoznam použitých skratiek a symbolov	61
Zoznam príloh	62

ÚVOD

Proces rekonštrukcie vozovky spočíva v prevedení fyzického tvaru vozovky do formy priestorového polygonálneho modelu. Začína nasnímaním vozovky vhodnou skenovacou metódou, pričom dáta, ktoré sú výstupom zo skenovania sú zvyčajne vyjadrené formou mračna bodov. Častokrát sú však sprevádzané šumom, ktorý je snaha eliminovať alebo prinajmenšom do značnej miery redukovať. Najvýhodnejším prípadom je pokiaľ je šum redukovateľný ešte v rámci procesu skenovania, napríklad zmenou / optimalizáciou skenovacích podmienok, skenovacieho prístroja alebo prípadne celkovou zmenou skenovacej metodiky. Pokiaľ to však nie je dosiahnuteľné, je možné aplikovať na mračno bodov niektorý z rady odšumovacích algoritmov, ktoré budú predstavené v tejto práci.

Pre získanie povrchu z bodov sa vykonáva krok triangulácie, tj. vytvorenie polygonálnej siete. Táto vytvorená sieť však pokrýva iba miesta ktoré boli nasnímané – v tomto prípade povrch trate. Pokiaľ je to účelné, je možné vyplniť nezosnímané miesta okolo trate buďto na základe nejakej všeobecnej siete za účelom vizualizácie alebo aj za pomoci výškových máp, pokiaľ je žiaduce aby okolitý terén zodpovedal čo najviac skutočnosti. Takto rekonštruovaná trať môže byť potom použitá pre rôzne účely, napríklad pre získanie parametrov trate ako sú vzdialenosti, sklon apod. (reverzné inžinierstvo).

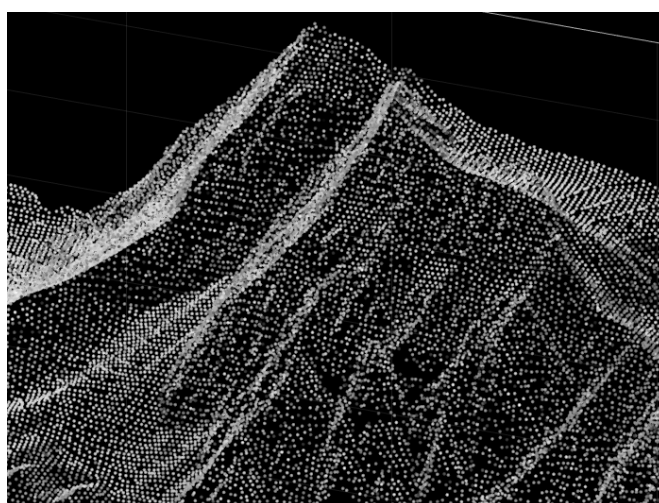
1 ZÁKLADY SKENOVANIA OBJEKTU

Skenovanie priestorového objektu je proces, ktorý má za úlohu získať informácie o fyzickom objekte, ako sú jeho rozmery, povrch, farba. Skenovaním sa získavajú najmä vlastnosti objektu z jeho povrchu, avšak niektoré metódy skenovania sú schopné nahliadnuť aj do vnútra objektu bez potreby deštruktívneho testovania. Praktické využitie nachádza 3D skenovanie hlavne v problematikách výskumu objektov, kontroly kvality a reverzného inžinierstva a to v mnohých odvetviach ako je strojný, stavebný a biomedicínske inžinierstvo, zábavný priemysel (filmy, hry, virtuálna realita) a ďalších. Proces skenovania začína zaznamenaním dát z objektu, kedy je potrebné získať dostatočné množstvo dát, ktoré budú vstupom pre ich následné spracovanie. Kvalita zosnímania sa môže výrazne odlišovať v závislosti na kvalite získaných dát, ktorá môže byť daná napr. kvalitou snímacieho zariadenia, snímacou frekvenciou, metodikou snímania, vonkajšími vplyvmi apod. Získané dáta sa následne spracujú, pričom veľa krokov je možné automatizovať na základe znalosti skenovacích podmienok. Pre tento účel je možné využiť rôzne zaužívané algoritmy. Každá z metód skenovania so sebou prináša radu výhod a nevýhod, prípadne obmedzení ktoré je potrebné vziať do úvahy pri určovaní metodiky skenovacieho procesu.

1.1 ZÁKLADNÁ TERMINOLÓGIA

MRAČNO BODOV

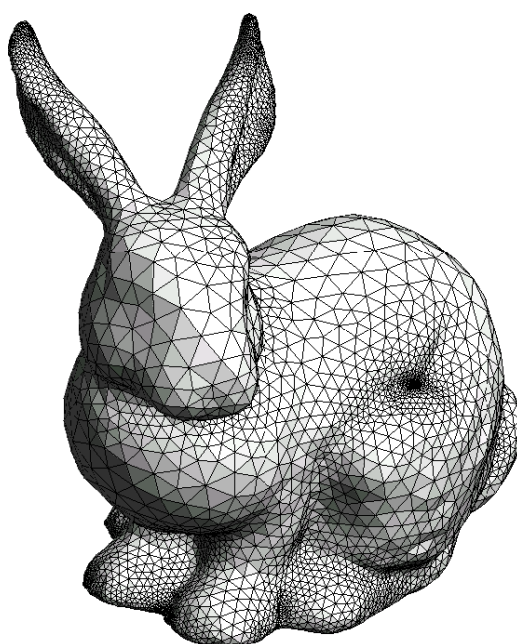
Súbor bodov v priestore, definovaných zväčša kartézskymi súradnicami. Body môžu byť nositeľom aj iných informácií ako len ich umiestnenia, môžu obsahovať taktiež informáciu o farbe, teplote či inej vlastnosti. Po vytvorení mračna nasleduje štandardne jeho úprava do vhodnejšej podoby, napr. 3D modelu, nakoľko samotné mračno má (vzhľadom na spoľahlivé vyhodnotenie) často relatívne obmedzenú praktickú použiteľnosť nakoľko sa zväčša jedná o veľký súbor neusporiadaných dát, niekedy obsahujúcich taktiež nejakú úroveň šumu. Naskenované body vyjadrujú zvyčajne povrch skenovaného objektu, avšak v niektorých zvláštnych prípadoch, ktoré vyžadujú poznať aj vnútro objektu môžu mračná vyjadrovať tiež celý objem objektu, vrátane jeho vnútorného usporiadania.



Obr. 1 Mračno bodov – terén [13]

POLYGONÁLNA SIEŤ

Plošný alebo priestorový súbor spolu spojených polygónov zložených z vrcholov, hrán a stien ktoré spolu vyjadrujú určitú plochu (na rozdiel od objemovej siete, ktorá je zložená z priestorových prvkov vyjadrujúcich objem). Všetky vrcholy jednotlivých stien musia byť vždy koplanárne, tzn. nemôže nastať, že napríklad vrcholy štvoruholníkového polygónu nebudú ležať v jednej rovine. V takomto prípade by bolo nutné rozdeliť štvoruholník na dve trojuholníkové steny. Práve vďaka schopnosti vyjadriť prakticky akýkoľvek tvar je hlavne pre nepravidelné tvary najrozšírenejším typom polygonálnej siete sieť trojuholníková, kedy je povrch tvorený sieťou trojuholníkov ako je vidieť napríklad na obr. 2. Často je možné sa stretnúť s nerovnomernou veľkosťou polygónov, ako je to zreteľné aj na danom obrázku, kde v miestach detailov je sieť jemnejšia ako v miestach, kde je detailov menej.



Obr. 2 Priestorová trojuholníková sieť [14]

VRCHOL

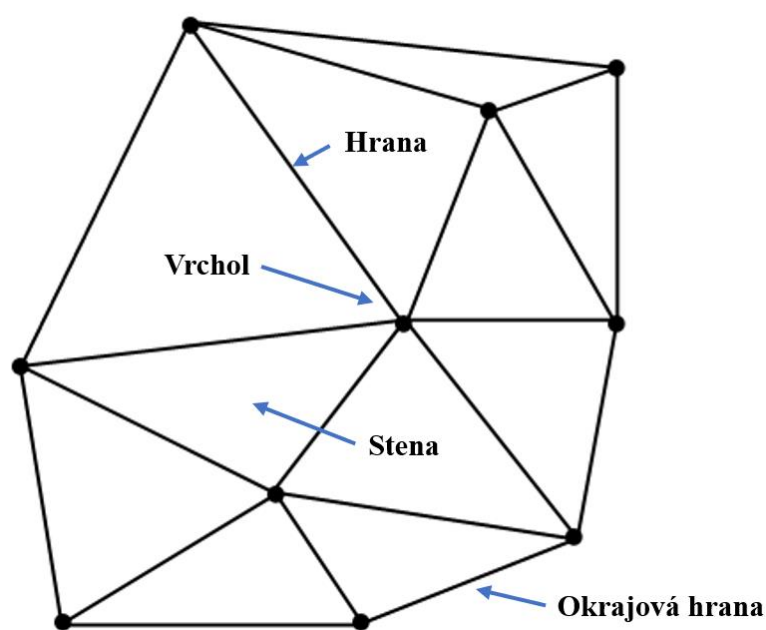
Bod v súbore polygonálnej siete. V kontexte spracovania mračna bodov sa po triangulácii práve jednotlivé body mračna stávajú vrcholmi siete.

HRANA

Priame spojenie medzi dvoma vrcholmi polygonálnej siete. Hrana je okrajová pokiaľ patrí iba jednému polygónu.

STENA

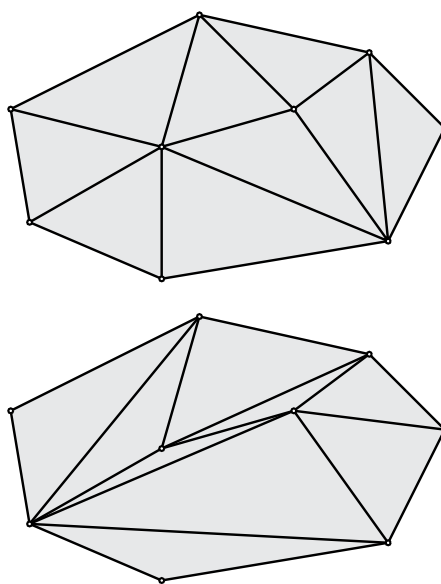
Množina vzájomne prepojených hrán tvoriacich uzavretý okruh. V prípade trojuholníkovej siete je stena definovaná tromi hranami, v prípade štvorcovej štyrmi apod.



Obr. 3 Základné stavebné elementy polygonálnej siete [15]

TRIANGULÁCIA

Usporiadanie dát do formy trojuholníkovej siete, v prípade spracovania mračna bodov sa jedná o vhodné prepojenie bodov medzi sebou hranami. V závislosti od metódy triangulácie je táto vhodnosť posudzovaná každý raz iným spôsobom, cieľom môže byť napríklad dosiahnuť čo najkratšie strany trojuholníkov spájaním vždy iba najbližších bodov. Iným cieľom môže byť napríklad dosiahnutie čo najväčších minimálnych uhlov v trojuholníkoch, aby sa predišlo vytvoreniu trojuholníkov s príliš ostrými uhlami. [3]



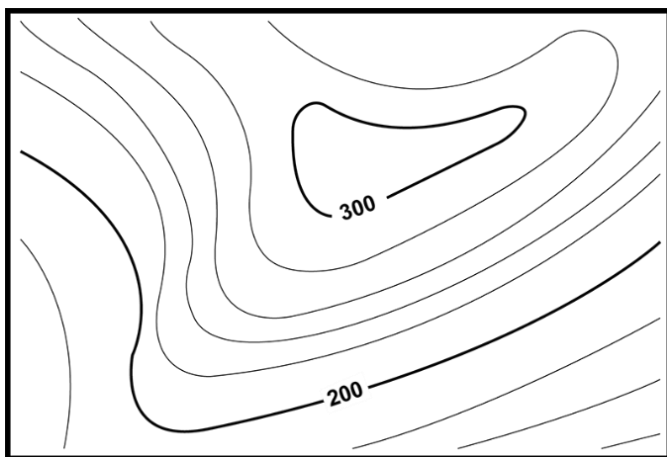
Obr. 4 Odlišné výsledky v závislosti na použitej metóde triangulácie identickej množiny bodov [16]

REKONŠTRUKCIA

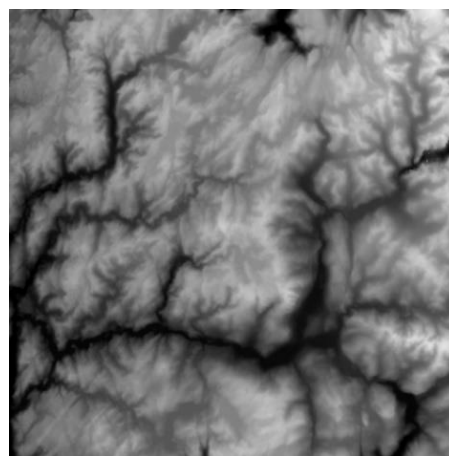
Proces vytvorenia digitálneho vyjadrenia skutočného fyzického objektu na základe naskenovaných dát.

DIGITÁLNY MODEL TERÉNU

Trojrozmerné vyjadrenie reliéfu povrchu (zvyčajne zemského) zostavené pomocou výšok. Najčastejšie je možné sa stretnúť s vyjadrením pomocou vrstevníc (obr. 5) alebo výškovej mapy (obr. 6). Vrstevnice sú izočiar, ktoré spájajú miesta s rovnakou nadmorskou výškou. Kvôli prehľadnosti sú zobrazené v určitých krokoch podľa výšky. Alternatívnym vyjadrením modelu terénu sú výškové body, kde každý bod nesie informáciu o konkrétnej výške, pričom tieto body sú rovnomerne rozmiestnené s určitou hustotou. Spolu tieto body tvoria výškovú mapu. V obidvoch prípadoch vyjadrenia však dáta nepokrývajú spojitě všetky miesta, tj. ak je potreba zistiť výšku bodu v nedefinovanom mieste, je nutné použiť vhodný spôsob interpolácie.



Obr. 5 Vrstevnicová mapa [17]



Obr. 6 Výšková mapa [18]

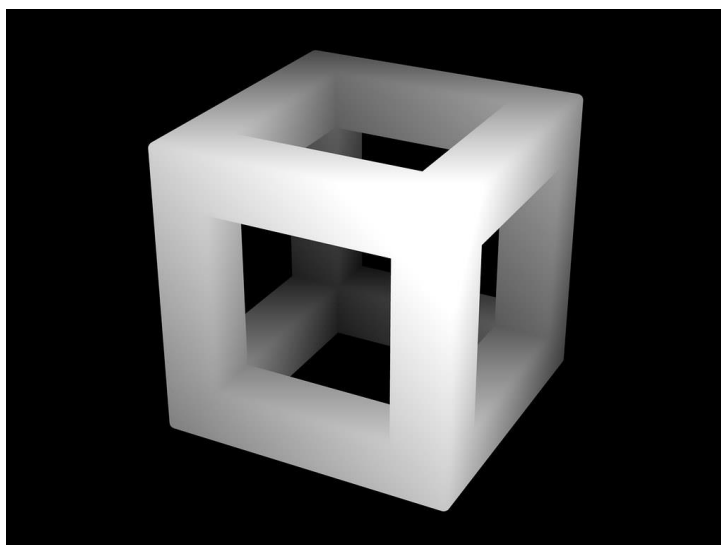
VÝŠKOVÁ MAPA

Sieť bodov, kde každý bod nesie informáciu o jeho výške. V prípade reliéfu je výšková mapa často prezentovaná vo forme grayscale obrázku (vid' obr. 6), v tomto prípade je výška bodu v každom mieste (pixeli) vyjadrená farbou pixelu – odtieňom sivej farby. Škála každého pixelu siaha od čiernej (hodnota 0) do bielej (hodnota 1), stupeň odtieňu sivej potom určuje výškovú hodnotu. Pre koreláciu so skutočnými nadmorskými výškami je taktiež nutné vedieť, akej výške zodpovedá čierny pixel a akej biely pixel, prípadne jednu z výšok v kombinácii s výškovým rozlíšením. Podrobnosť tejto mapy je daná dvoma ukazovateľmi – ako veľkú oblasť predstavuje jeden pixel (plošné rozlíšenie) a rozlíšenie farby pixelu (výškové rozlíšenie). Počet jednotlivých odtieňov, ktoré môže každý pixel v obrázku vyjadriť je daný bitovou hĺbkou obrázku. Čím je bitová hĺbka vyššia, tým viac odtieňov farby je možné zobrazit'. Nakoľko jeden bit je schopný niesť dva rôzne elementárne údaje, je napríklad pri dvoch bitoch možné vyjadriť 4 odtiene (00 – čierna, 01 – tmavošedá, 10 – svetlošedá, 11 - biela). Všeobecne n bitov sú schopné niesť 2^n jedinečných odtieňov. Najčastejšie sa stretávame s bitovou hĺbkou pri grayscale obrázkoch výškových máp okolo 8 – 16 bit, čo zodpovedá 256 – 65 536 odtieňom.

HĽBKOVÁ MAPA

Sieť bodov, kde každý bod nesie informáciu o jeho vzdialenosti od snímacieho prístroja v momente snímania. Často býva výstupom z laserového skenovania. Vo vizualizácii na obr. 7

je znázornená hĺbková mapa podobným spôsobom ako výšková mapa – v odtieňoch sivej farby od čiernej po bielu, avšak tentokrát farba neodpovedá výške objektu ale vzdialenosti daného miesta od snímacieho prístroja: čierna – najďalej, biela – najbližšie.



Obr. 7 Vizualizácia hĺbkovej mapy [19]

STROM (DÁTOVÁ ŠTRUKTÚRA)

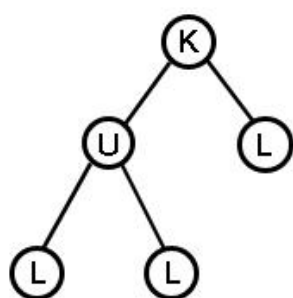
Strom je nelineárnou dátovou štruktúrou usporiadajúcou dáta do štruktúry pripomínajúcej strom. Skladá sa z uzlov vzájomne prepojených vetvami. Najvyšší uzol, tj. uzol ktorý ako jediný nemá rodiča, nazývame koreňom. Uzly bez potomkov potom nazývame listy (vid' obr. 8) V strome smie byť iba jeden koreň a každý uzol okrem koreňového uzlu musí mať práve jedného rodiča. Počet potomkov pre každý uzol nie je obmedzený, avšak pokiaľ má každý uzol maximálne dvoch potomkov, hovoríme o binárnom strome. Strom môže byť:

1. usporiadaný
2. neusporiadaný

V usporiadanom strome sú potomkovia uzlov zoradení, tj. existuje „prvý“ potomok a „posledný“ potomok. V neusporiadanom strome sú všetci potomkovia rovnocenní a neurčuje sa medzi nimi poradie.

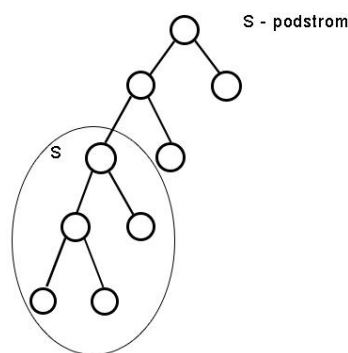
Podstrom je časť stromu, ktorá obsahuje jeden zvolený uzol a všetkých jeho potomkov, vid' obr. 9.

Úrovňami nazývame „generácie“ stromu, pričom koreňový uzol je na úrovni 0. Pokiaľ je uzol napríklad na 2. úrovni, je potomkom potomka koreňového uzlu. Najefektívnejším usporiadaním dát do stromu je také usporiadanie, ktoré obsahuje najmenší počet úrovní.



Obr. 8 Hlavné prvky stromu [20]

K - kořen
U - uzel
L - list



Obr. 9 Podstrom [21]

1.2 METÓDY SKENOVANIA

Všeobecne je možné rozdeliť metódy skenovania objektov na kontaktné a bezkontaktné. Kontaktné metódy využívajú pre skenovanie fyzický kontakt skenovacieho prístroja so skenovaným objektom, zatiaľ čo bezkontaktné metódy pracujú bez priamej fyzickej interakcie. [4]

Druhým spôsobom kategorizovania skenovacích metód je rozdelenie z hľadiska interakcie s objektom na aktívne a pasívne metódy. Pri aktívnych dochádza k určitej forme interakcie prístroja na skenovanie so skenovaným objektom. Môže sa jednať napríklad o získavanie hĺbkových máp pomocou rôznych typov diaľkometerov, napríklad laserového diaľkomeru. Pasívne metódy, na druhej strane, so skenovaným objektom nijak neinteragujú, pracujú len s informáciami ktoré je možné získať priamo z daného prostredia. [4]

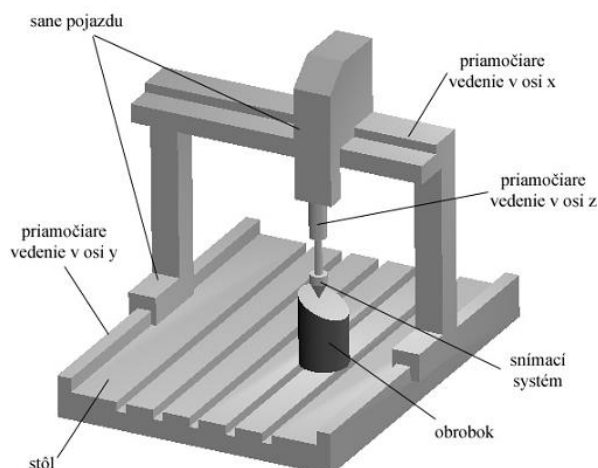
Vzhľadom na to, že kontaktné metódy z princípu podmieňujú, že sa jedná o aktívne metódy je možné všeobecne rozdeliť metódy skenovania na tri samostatné kategórie: kontaktné metódy, bezkontaktné aktívne a bezkontaktné pasívne.

1.2.1 KONTAKTNÉ METÓDY

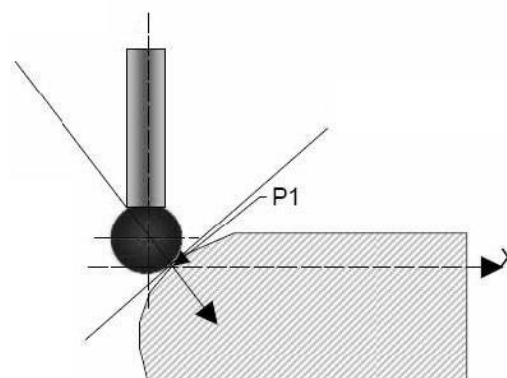
Skenovací prístroj získava informácie o objekte za pomoci jeho priameho kontaktu so skenovaným objektom. Kvôli tomu nachádzajú tieto metódy využitie najmä v pri menších a menej komplexných objektoch. Taktiež sú vhodné pre priehľadné alebo vysoko reflektívne predmety, pri ktorých metódy založené na zachytávaní svetla zaostávajú. Taktiež bývajú tieto metódy presnejšie ako metódy bezkontaktné. [4]

MERANIE POMOCOU CMM

Príkladom prístroja využívajúceho kontaktnú metódu snímania je súradnicový merací stroj – CMM (z angl. Coordinate Measuring Machine). Meranie pomocou CMM je uskutočňované pomocou meracej hlavice s meracím hrotom, ktorý, akonáhle sa dotkne skenovaného objektu, zaznamená súradnice hrotu. Takto prechádza skenovacia hlava celým objektom a zaznamenáva súradnice každého prejdeneho miesta. Výstupom z takéhoto zariadenia je priamo mračno bodov definovaných svojimi súradnicami. CMM nachádzajú okrem skenovania objektov za účelom rekonštrukcie 3D siete využitie aj v procese výroby pre kontrolu kvality výrobného procesu. [4]



Obr. 10 Všeobecná koncepcia CMM [22]



Obr. 11 Merací hrot prístroja CMM [22]

V niektorých prípadoch však nie je účelné použiť kontaktné metódy, či už z hľadiska samotnej metódy snímania, ktorá vyžaduje fyzický kontakt, tj. nie je vhodná pre objekty ktorých tvar môže byť skenovacou hlavou pozmenený alebo poškodený alebo aj výrazne nižšej rýchlosti snímania v porovnaní s bezkontaktnými metódami, hlavne pri skúmaní väčších alebo zložitejších objektov. [4]

1.2.2 BEZKONTAKTNÉ AKTÍVNE METÓDY

Metódy, ktoré sa nezakladajú na kontakte prístroja so skenovaným objektom ale zároveň s ním určitým spôsobom interagujú nazývame bezkontaktné aktívne metódy. Interakciu môže byť napríklad vyslanie lúču na skenovaný objekt a následné zachytenie odrazeného lúču senzorom. Jedným z takýchto prístrojov je napríklad laserový skener, ktorý je hojne využívaný aj pri skenovaní tratí. Nie je však jediným prístrojom pre bezkontaktný aktívny spôsob skenovania. Skenovacie prístroje môžu využívať veľké spektrum vlnových dĺžok – od ionizujúceho žiarenia – röntgenový prístroj až po rádiové vlny – radar, prípadne aj zvukové vlny – sonar.

LASEROVÉ SKENOVANIE

Jednou z najrozšírenejších metód skenovania je vďaka jej praktickosti a rýchlosti použitie laserového skeneru. V závislosti na jeho konštrukcii rozlišujeme Time-of-Flight skener, triangulačný skener a phase-shift skener. [4]

Time-of-flight, ako už názov napovedá je založený na vyslaní lúča a odmeraní času, koľko trvá lúču aby sa vrátil naspäť (viď obr. 12). Nakoľko je rýchlosť svetla známa, pomocou odmeraného času sa jednoducho vypočíta vzdialenosť miesta, na ktoré je laserový skener namierený [4]:

$$d = \frac{c * t}{2}, \quad (1)$$

kde d – vzdialenosť bodu od snímača

c – rýchlosť svetla

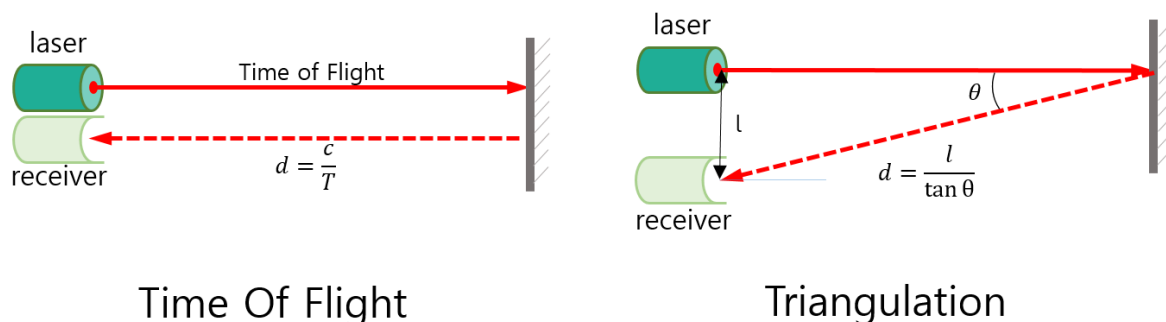
t – nameraný čas – čas od vyslania lúča po opätovné zachytenie odrazeného lúča

Takto sa pomocou skeneru vyšetrí všetky miesta aktuálneho zorného poľa skeneru, čím sa získa hĺbková mapa. Kde však metóda môže naraziť na problém je pri vyšetrovaní hrán, kedy lúč dopadne tak, že jedna polovica dopadá na bližší objekt a druhá polovica na vzdialenejší objekt. V tom prípade je vypočítaný priemer z týchto vzdialeností, tzn. bod je umiestnený na nesprávne miesto (tento efekt je však možné utlmiť pomocou softwaru). Taktiež, nakoľko musí byť čas meraný veľmi presne (rádovo v pikosekundách), býva dosiahnutá presnosť horšia ako u metódy triangulácie – rádovo v milimetroch. [4]

Skener ktorý patrí do kategórie phase-shift skenerov využíva pre posúdenie vzdialenosti fázový posun medzi odoslanou vlnou a prijatou. Princíp funkcie je však podobný ako u Time-of-Flight prístroja. [4]

Alternatívnym spôsobom získavania hĺbkovej mapy je skenovanie pomocou triangulácie, kedy prístroj vyšle laserový lúč a senzor zaznamená polohu bodu lúča dopadnutého na vyšetřovaný objekt (viď obr. 12). Podľa vzdialenosti miesta dopadu lúča sa lúč zobrazí na iných miestach. Tento senzor nesmie byť veľmi blízko zdroju lúča, čím je totiž bližšie, tým je ťažšie presne odhadnúť vzdialenosť (malá variácia polohy so zväčšujúcou sa vzdialenosťou). Nakoľko je známa vzdialenosť medzi zdrojom laserového lúča a senzorom a poloha dopadnutého lúču vzhľadom na senzor je zaznamenaná, je trojuholník plne definovaný a je možné vypočítať vzdialenosť snímaného miesta od zdroju svetla (prípadne aj od senzoru). V praxi je často namiesto zdroju bodového svetla používaný pruhový zdroj svetla, čím je možné v jednom kroku získať vzdialenosť veľkého počtu miest, na ktoré pruh svetla dopadne. [4]

Na rozdiel od ToF snímání je snímání pomocou triangulácie presnejšie – rádovo v desiatkach mikrometrov, na druhej strane však neponúkajú tak veľký dosah ako prístroje ToF – efektívne je možné snímať rádovo v metroch, zatiaľ čo pri ToF je možné snímať objekty vzdialené až vo vzdialenosti niekoľko kilometrov. [4]



Obr. 12 Time-of-Flight a triangulačné snímání [23]

1.2.3 BEZKONTAKTNÉ PASÍVNE METÓDY

Metódy, ktoré so skenovaným objektom nijak priamo neinteragujú nazývame bezkontaktné pasívne metódy. Tieto metódy sa zakladajú na získaní údajov, ktoré sú v danom momente dostupné bez akéhokolvek zásahu – na rozdiel od aktívnych metód, ktoré okrem snímača disponujú aj určitým vysielačom, zariadenia pre pasívne metódy snímání potrebujú pre svoju funkciu iba snímač. Zväčša sa využíva viditeľného svetla, avšak je možné použiť aj žiarenia infračervených vlnových dĺžok. Hlavným predstaviteľom kategórie pasívnych bezkontaktných metód je snímání pomocou fotogrametrie. K tomuto typu snímání je často potrebný iba dostatočne kvalitný fotoaparát. Aj vďaka svojej prostriedkovej nenáročnosti často nachádzajú

využitie pri amatérskom 3D skenovaní, avšak pri dostatočnej presnosti snímania ponúka aj profesionálne využitie. [4]

FOTOGRAMETRIA

Fotogrametria je metóda ktorá využíva na snímanie prostredia fotografický senzor. Z jednej snímky však nie je možné získať informáciu o vzdialenosti objektov na snímke. Pre zostrojenie trojrozmerného vyjadrenia je možné túto hĺbkovú informáciu získať zachytením viacerých odlišných snímok rovnakého objektu. Odlišnosť môže byť daná napríklad nasvietením objektu z rôznych strán, kedy sa pozoruje odraz svetla od rôznych plôch, čím je možné aproximovať normálové vektory týchto viditeľných plôch alebo nasnímaním objektu z rôznych uhlov a následné spojenie snímok za pomoci detekcie rovnakých bodov v odlišných snímkach.

Rozšírením prístroja o druhý snímač orientovaný rovnakým smerom sa získa metóda stereofotogrametrie. Každá z kamier sníma vplyvom vzájomného odsadenia trochu odlišný obraz, pričom pomocou tejto odlišnosti je možné odhadnúť vzdialenosť každého z objektov – čím je ich umiestnenie na rôznych snímačoch odlišnejšie, tým je objekt bližšie. [4]

1.3 METODIKA SPRACOVANIA DÁT

V každej z týchto metód snímania je možné sa v určitej fáze spracovania dát stretnúť s problematikou spracovania súboru bodov. Vo väčšine prípadov nie sú prvotné neupravené súbory bodov priamo použiteľné a je často nutné ich vhodným spôsobom spracovať. [5] Ciele ktorých je snaha týmto spracovaním dosiahnuť sa môžu od seba mierne líšiť, avšak často je možné medzi jednotlivými metodikami spracovania nájsť určitú podobnosť. V jednom prípade môže byť hlavným dôvodom odstránenie nežiaducích bodov, v druhom napríklad o zníženie výpočtového času pri zachovaní detailov.

Pre tieto účely existuje veľké množstvo algoritmov, ktoré môžu byť nápomocné pri jednotlivých krokoch úpravy. Niektoré metódy pracujú priamo so súborom bodov, niektoré vyžadujú prvotne vytvorenie polygonálnej siete (najčastejšie trojuholníkovej – triangulácia), ktorá je potom spracovávaná.

FÁZY

Spracovanie mračna bodov sa riadi krokmi ktoré sú usporiadané do určitých fáz. V závislosti od metódy získania dát a získanej a požadovanej kvality / formy dát môžu byť niektoré kroky na želanie vynechané, prípadne môžu byť potrebné aj iné úpravy, avšak vo všeobecnosti sa jedná o tieto fázy [5]:

1. Preprocessing mračna bodov
2. Rekonštrukcia siete
3. Segmentácia mračna bodov
4. Modelovanie mračna bodov

Preprocessing, alebo nejaké pred-spracovanie mračna bodov má za úlohu upraviť množinu bodov do takej podoby, aby čo najlepšie vystihovala skutočnosť a zároveň sa s týmto súborom čo najjednoduchšie pracovalo. Zahŕňa v sebe zvyčajne tieto kroky [5]:

1. Registrácia

2. Filtrovanie šumu a odstránenie extrémnych hodnôt
3. Down-sampling

Postup sa však neobmedzuje iba na tieto tri kroky, požadované môžu byť taktiež ďalšie kroky, napríklad pokiaľ je známe, že prístroj zaznamenáva hodnoty odchýlené od skutočných hodnôt, je možné zistiť korekčné posuvy, prípadne súčinitele a môžu byť takisto aplikované v tejto fáze.

1.3.1 PREPROCESSING – REGISTRÁCIA

Registrácia je nutným krokom pokiaľ sú vstupom do spracovania viaceré súbory bodov, ktoré sú vyjadrené vzhľadom k rôznym súradným systémom a je snaha obdržať v konečnom dôsledku jeden súbor dát. Zvyčajne sa používa tzv. ICP algoritmus (z angl. Iterative Closest Point), ktorý pomocou určitého počtu iterácií dokáže spojiť dva súbory bodov odhadnutím transformačnej matice pomocou minimalizovania vzdialenosti medzi korešpondujúcimi bodmi v oboch množinách.

1.3.2 PREPROCESSING – FILTROVANIE ŠUMU A ODSTRÁNENIE EXTRÉMNYCH HODNÔT

Šumom nazývame body, ktoré sú obsiahnuté v naskenovanom mračne bodov ale ich umiestnenie nie je správne, čo môže byť ovplyvnené napr. presnosťou / kvalitou skenovacieho prístroja / obmedzeniami skenovacieho procesu, metodikou skenovania, vonkajšími podmienkami apod. Outliermi alebo extrémnymi bodmi nazývame všetky body, ktoré sú obsiahnuté v mračne bodov, avšak nespádajú do spektra pozorovania, tj. nevyjadrujú objekt záujmu. Obidve tieto množiny bodov sú objektom spracovania. Pre odstránenie šumu sa používajú odšumovacie algoritmy. Pre odstránenie extrémnych hodnôt sa používajú algoritmy na hľadanie a elimináciu týchto bodov.

Existuje veľmi široké spektrum odšumovacích algoritmov, preto budú pre jednoduchosť ilustrované iba algoritmy použité v tejto práci (prípadne s nimi úzko súvisiace).

BILATERÁLNE FILTROVANIE

Odšumovanie pomocou bilaterálneho filtra sa zaradzuje medzi metódy odšumovania založené na susedstve bodov, tzn. posun každého bodu v záujme odšumenia sa realizuje na základe umiestnenia najbližších susedných bodov daného bodu. Aj keď je daný algoritmus nazvaný filtrovaním, bilaterálny filter body neodstraňuje ani nepridáva.

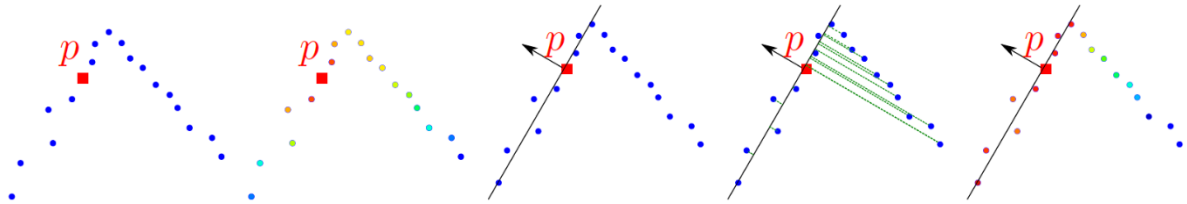
Bilaterálny filter však nebol od začiatku určený pre odšumovanie mračien bodov. Jeho prvým a doteraz hlavným cieľom je odšumovanie digitálnych fotografií, kde sa modifikujú farby pixelov obrázku na základe susedstva, ktoré tvoria okolité pixely. Blízkosť pixelov však nie je určená iba ich absolútnou vzdialenosťou od odšumovaného pixelu, ale aj ich vzájomnou podobnosťou – preto má každý okolitý pixel svoju váhu, ktorou závaží pri určovaní novej farby odšumovaného pixelu zloženú z dvoch komponentov – vyššiu váhu budú mať pixely ktoré sú aj blízko umiestnením, aj podobnosťou. Tým, že je prihliadané na dva faktory získalo toto filtrovanie názov bilaterálne. Čím však získal na popularite je vlastnosť dobre zachovávať ostré prechody, čím sa obrázky zjemňujú bez zjemnenia hlavných prvkov obrázku. Metóda je štandardne neiteratívna, tj. prebieha v jednom kroku a variácia výsledkov je daná nastavením parametrov. [6]

Aplikácia bilaterálneho filtra na súbor bodov je vykonaná analogicky, pričom namiesto farby pixelov je modifikované umiestnenie bodov. V kontexte spracovania mračien bodov prihliada

algoritmus bilaterálneho filtrovania pri určovaní novej polohy bodu na dva faktory [7]:

1. Absolútna vzdialenosť susedného bodu k odšumovanému bodu
2. Vzdialenosť susedného bodu k regresnej ploche odšumovaného bodu a jeho susedov

Pomocou týchto ukazovateľov sa určí váha každého susedného bodu (viď obr. 13). S menšou vzdialenosťou sa váha bodu zvyšuje. Avšak nakoľko je do úvahy braná aj vzdialenosť k regresnej ploche, tak algoritmus si vie dobre poradiť so zachovávaním hrán. [7]



Obr. 13 Bilaterálne filtrovanie. V poradí zľava doprava: 1. Odšumovaný bod a jeho susedné body 2. Váha susedných bodov podľa absolútnej vzdialenosti 3. Preloženie regresnej plochy 4. Určenie vzdialenosti od regresnej plochy 5. Kombinovaná váha susedných bodov [7]

Algoritmus bilaterálneho filtrovania sa dá zapísať nasledovne [7]:

$$[P_d] = \text{bilateral}(P, \sigma_d, \sigma_n, N_r(P), n_p)$$

$$\text{sum}_w = 0$$

$$\delta_p = 0$$

for $Q \in N_r(P)$

$$d_d = \|Q - P\|$$

$$d_n = \langle Q - P, n_p \rangle$$

$$w = e^{-\frac{d_d^2}{2\sigma_d^2}} * e^{-\frac{d_n^2}{2\sigma_n^2}}$$

$$\delta_p = \delta_p + w * d_n$$

$$\text{sum}_w = \text{sum}_w + w$$

end

$$P_d = P + \frac{\delta_p}{\text{sum}_w} * n_p$$

kde: P – odšumovaný bod,

P_d – odšumený bod,

Q – bod z množiny susedných bodov k bodu P ,

$N_r(P)$ – množina susedných bodov k bodu P ,

n_p – normálový vektor regresnej plochy v bode P ,

σ_d – Gaussova váha pre euklidovskú (geometrickú) vzdialenosť,

σ_n – Gaussova váha pre vzdialenosť k regresnej ploche,

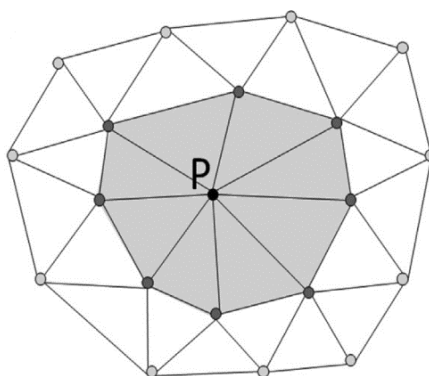
d_d – euklidovská (geometrická) vzdialenosť bodu Q k bodu P,

d_n – vzdialenosť bodu Q k regresnej ploche v bode P.

Pred samotným odšumovaním je teda nutné nájsť pre každý odšumovaný bod mračna bodov jeho susedné body a podľa tejto informácie aj aproximáciu normálového vektora v každom odšumovanom bode, čo je vstupom do algoritmu bilaterálneho filtrovania. Tieto metódy, vrátane plného postupu prístupu užívajúceho bilaterálny filter, budú popísané v kapitole 3.2.

LAPLACOVO ODŠUMOVANIE

Laplacovo odšumovanie a jeho rozšírené varianty sa radia medzi najpoužívannejšie metódy odšumovania sietí. [8] Na rozdiel od bilaterálneho filtrovania je Laplacovo odšumovanie založené na odšumovaní polygonálnej siete a nie priamo mračna bodov. Preto pred samotným odšumovaním sa vykoná triangulácia a následne algoritmus pracuje s touto sieťou. Laplacova metóda odšumovania je iteratívna, tj. je opakovaná po určitý počet iterácií N, čím sa tiež odlišuje od bilaterálneho filtrovania, ktoré je jednokrokové. Spoločnou vlastnosťou tejto metódy a metódy bilaterálneho filtrovania je, že táto metóda je taktiež založená na susedstve bodov – vrcholov (v tomto prípade hovoríme o polygonálnej sieti, takže bodmi rozumieme vrcholy siete), tj. práve okolité vrcholy určujú smer posunu odšumovaného vrcholu. V tomto prípade sú však uvažované iba susedné vrcholy 1. rádu, tj. vrcholy ktoré zdieľajú s odšumovaným vrcholom hranu (viď obr. 14). Magnitúda posunu je potom určená odšumovacím parametrom λ .



Obr. 14 Bod P a jeho susedné vrcholy 1. rádu [24]

Laplacovo odšumovanie, v najjednoduchšej forme, posúva každý odšumovaný bod v smere ťažiska jeho susedných bodov 1. rádu [8]:

$$P_d = P + \lambda * U(P), \quad (2)$$

$$U(P) = \frac{1}{\sum_i w_i} \sum_i w_i * Q_i - P, \quad (3)$$

kde: P – odšumovaný vrchol,

P_d – odšumený vrchol,

$U(P)$ – „dáždnikový“ operátor, vid' obr. 15,

λ – odšumovací parameter z intervalu $0 < \lambda < 1$,

Q_i – i-tý susedný vrchol vrcholu P ,

w_i – váha i-tého susedného vrcholu.

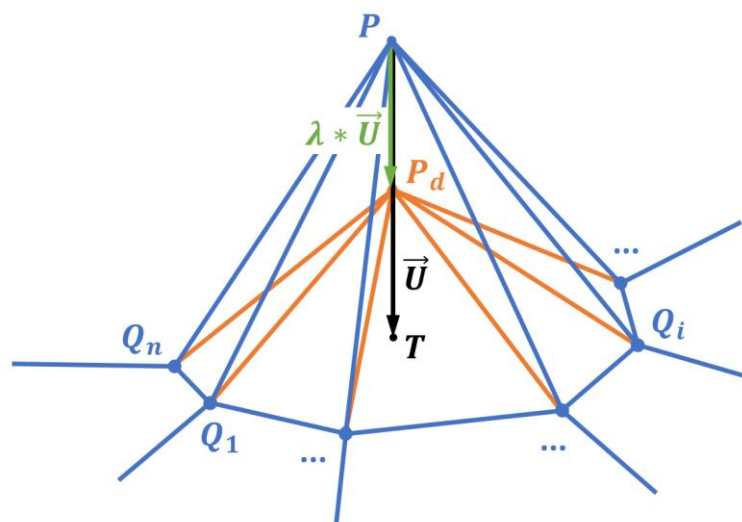
Váha môže byť definovaná rovnaká pre všetky susedné vrcholy [8]:

$$w_i = 1, \quad (4)$$

alebo taktiež je možné ju položiť rovnú obrátenej hodnote vzdialenosti daného susedného vrcholu od odšumovaného vrcholu [8]:

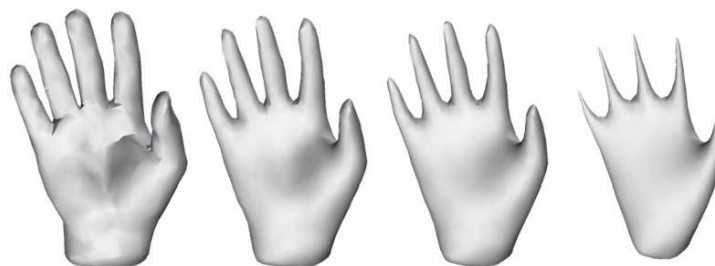
$$w_i = \frac{1}{\|P - Q_i\|}, \quad (5)$$

Je však nutné poznamenať, že pri zvolení rovnakej váhy všetkých bodov sa sieť zrovnomerňuje, zatiaľ čo s obrátenou hodnotou vzdialenosti sa sieť zhoršuje. [8]



Obr. 15 „Dáždnikový operátor“ – Laplacovo odšumovanie

V tejto základnej forme má však Laplacova metóda obmedzenú použiteľnosť, nakoľko takýto typ odšumovania má tendenciu zmršťovať sieť, čo je nežiaduci jav, vid' obr. 16. [9]

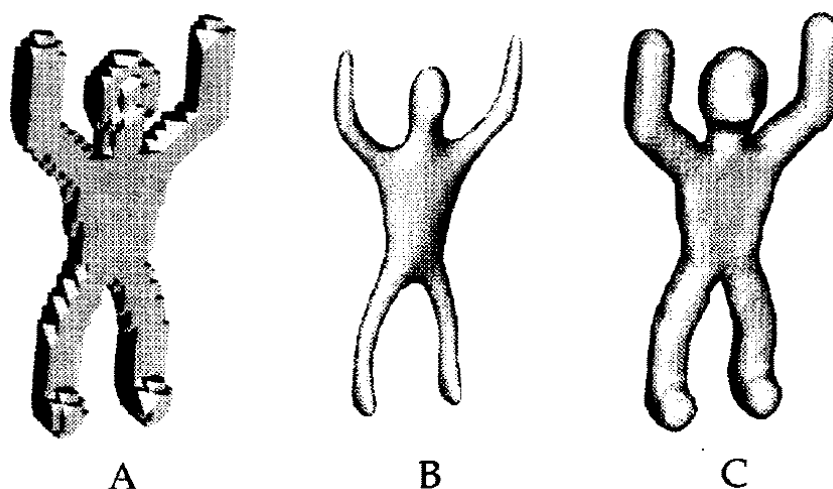


Obr. 16 Zmršťovanie siete s narastajúcim počtom iterácií (vľavo neodšumený model) [25]

Ukázalo sa, že pokiaľ sa v každej iterácii na súbor bodov ktorý už prešiel Laplacovým odšumovaním s parametrom λ , pridá druhý krok, kedy sa rovnakým princípom posunie každý bod, ale v opačnom smere s parametrom μ (pričom $\mu < 0$ a $|\mu| \geq \lambda$), tento efekt zmršťovania sa minimalizuje. [9] Podľa parametru μ sa takáto metóda nazýva [8]:

- a) bilaplaciálne odšumovanie (pokiaľ $|\mu| = \lambda$)
- b) Taubinovo odšumovanie (pokiaľ $|\mu| > \lambda$)

Práve Taubinovo odšumovanie dokáže najlepšie minimalizovať efekt zmršťovania siete, ako je to znázornené na obr. 17. [9]



Obr. 17 Minimalizovanie zmršťovania siete. A – východzí model. B – Laplacovo odšumovanie. C – Taubinovo odšumovanie [9]

Postup algoritmu potom vyzerá nasledovne:

```

[XYZd] = taubin(XYZ, λ, μ, N, np)
for n = 1:N
  for P ∈ XYZ
    for Q ∈ N(P)
      
$$U(P) = \frac{1}{\sum_i w_i} \sum_i w_i * Q_i - P$$

      
$$P_d = P + \lambda * U(P)$$

    end
  end
  XYZ = XYZd
  for P ∈ XYZ
    for Q ∈ N(P)
      
$$U(P) = \frac{1}{\sum_i w_i} \sum_i w_i * Q_i - P$$

      
$$P_d = P + \mu * U(P)$$

    end
  end
end

```

Plný postup aplikácie prístupu užívajúceho variantu Laplacoveho odšumovania, bude popísaný v kapitole 3.3.

1.3.3 PREPROCESSING – DOWN-SAMPLING

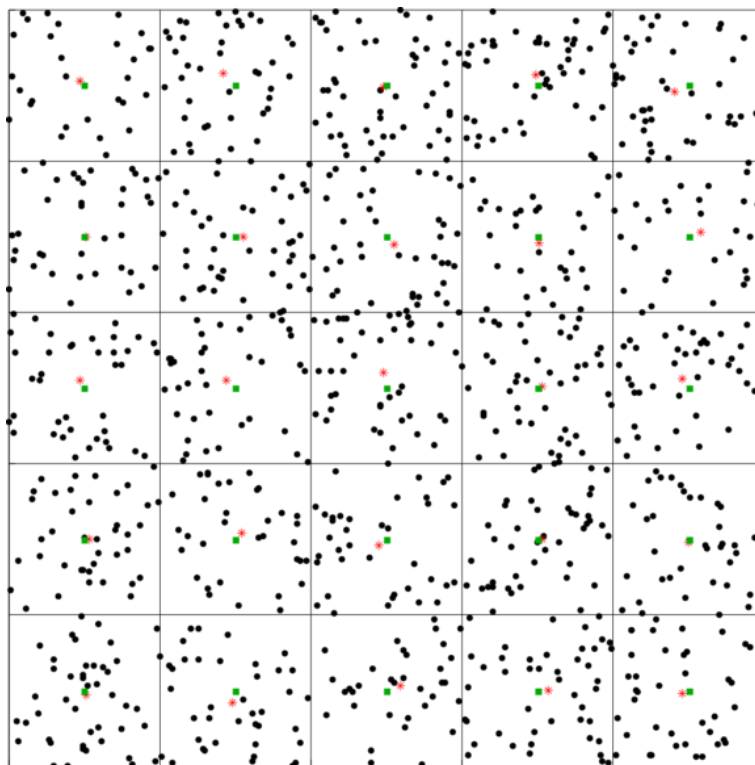
Pre zredukovanie výpočtového času a zároveň zrovnomenie hustoty mračna bodov je vhodné zredukovať mračno vhodným spôsobom pre získanie ľahšie spracovateľného dátového súboru pri čo najväčšom stupni zachovania pôvodných detailov.

FILTROVANIE POMOCOU VOXELOVEJ SIETE

Algoritmus redukuje množstvo bodov pomocou pomyslenej priestorovej mriežky – voxelovej siete, ktorú preloží cez mračno bodov. Mriežka môže byť rovnomerná – tj. všetky jednotlivé bunky sú rovnako veľké, prípadne lokálne zjemnená – napríklad v mieste výskytu detailov. V každom voxeli sa body prislúchajúce danému voxelu substituujú jedným bodom. Tento bod môže byť:

1. geometrickým ťažiskom všetkých bodov v danom voxeli
2. geometrickým ťažiskom daného voxelu

Prvý variant obnáša väčšiu presnosť nakoľko pri určovaní polohy substitučných bodov je prihliadané na polohu všetkých bodov v danom mračne a body sú nahradené miestom ich skutočného ťažiska. Druhá metóda je menej presná, avšak veľmi rýchla nakoľko jedinou informáciou pre umiestnenie substitučného bodu do bunky je, či daná bunka obsahuje aspoň jeden bod mračna. Taktiež produkuje dokonale rovnomernú sieť. Tieto dva prístupy je možné vidieť na obr. 18. Červené body vyjadrujú prvú variantu, tj. nahradenie ťažiskom bodov, zelené body vyjadrujú nahradenie ťažiskom daného voxelu.



Obr. 18 Dvojmerná voxelová sieť – nahradenie ťažiskom bodov a ťažiskom voxelov [26]

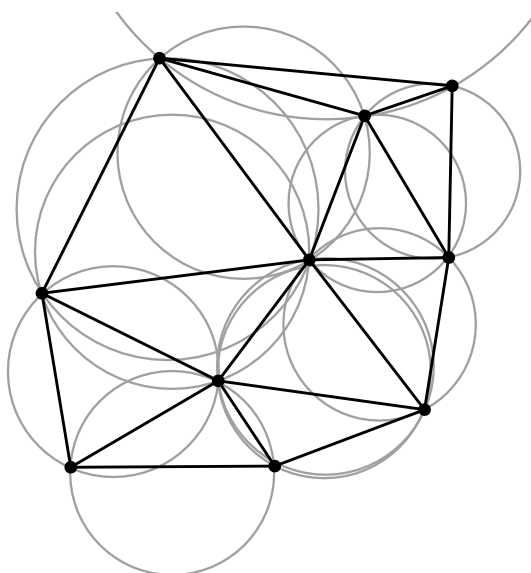
Okrem redukcie objemu dát má filtrovanie pomocou rovnomernej voxelovej mriežky za následok aj zrovnomenenie hustoty bodov, ktoré môže byť výhodné, prípadne aj potrebné pre viaceré algoritmy aj pre kvalitu siete, pokiaľ je rovnomernosť siete žiaducim efektom.

1.3.4 REKONŠTRUKCIA SIETE

Cieľom tohto kroku je vytvoriť sieťový model skúmaného objektu z mračna bodov, ktoré sa získa z predchádzajúceho kroku. Takto vytvorená sieť je jednoduchšia na vizualizáciu a môže byť v takejto podobe priamo použitá pre rôzne účely. Štandardne sa tento krok dosiahne pomocou triangulácie.

Všeobecne sa algoritmy triangulácie rozdeľujú na metódy založené na Delaunay triangulácii a metódy nezávislé od Delaunay triangulácie. [3]

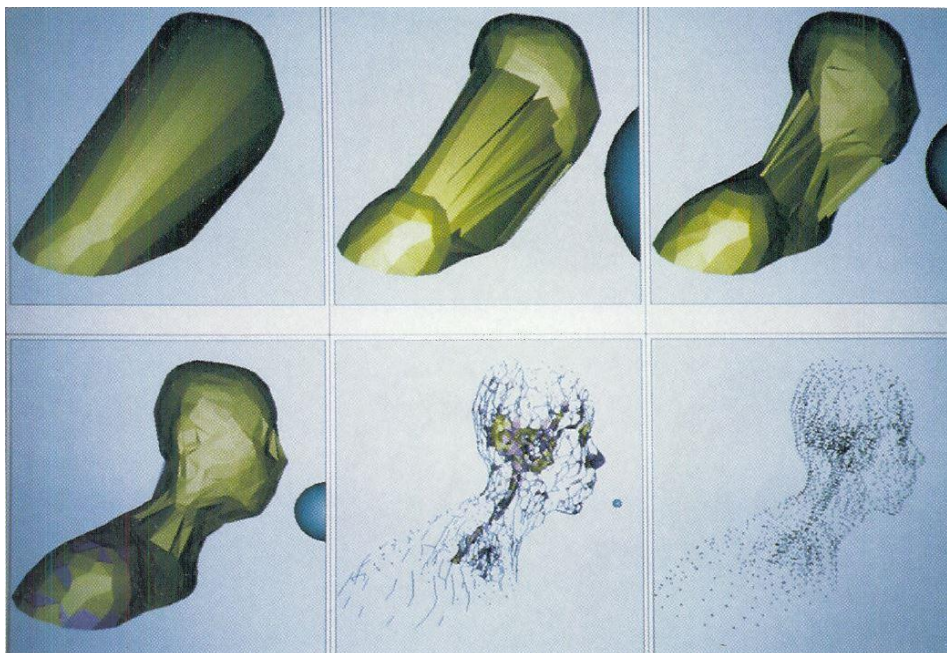
Jednou z charakteristických vlastností Delaunay triangulácie je, že pokiaľ sa zostrojí opísaná kružnica akémukoľvek polygónu siete, vnútro tejto kružnice nebude obsahovať žiaden iný bod ako iba body, ktoré zostrojujú daný polygón, vid' obr. 19. Taktiež Delaunay triangulácia má za cieľ maximalizáciu minimálnych uhlov trojuholníkov prítomných v sieti pre zabránenie výskytu príliš ostrých trojuholníkov, ktoré zvyčajne znižujú kvalitu siete. [3]



Obr. 19 Opísaná kružnica každého z trojuholníkov Delaunay triangulácie neobsahuje žiaden iný bod okrem bodov trojuholníka [15]

Nakoľko žiadna z metód nezávislých od Delaunay triangulácie popísaných v [3] by nebola vhodná na zadaný dátový súbor, je v tejto práci je použitá metóda, ktorá vychádza z Delaunay triangulácie – Alpha shapes. Výhodou okrem širokej použiteľnosti je taktiež rozšírenosť tejto metódy, čo naznačuje aj prítomnosť vstavanej funkcie v Matlabe práve na trianguláciu pomocou Alpha shapes.

Alpha shapes algoritmus je vlastne algoritmom, ktorý vytvorí Delaunay trianguláciu, na ktorej vykoná filtrovanie pomocou parametru α . Filtrovanie siete je možné si predstaviť ako prechod gule s polomerom α celým modelom, pričom podľa tohto polomeru sa nastavuje miera odstraňovania hrán triangulácie, vid' obr. 20. [10]



Obr. 20 Vplyv parametru (polomeru gule) alfa na filtráciu triangulácie [10]

1.3.5 SEGMENTÁCIA MRAČNA BODOV

Úlohou segmentácie mračna bodov je nájsť v mračne oblasti zodpovedajúce rôznym objektom a rozdeliť toto mračno na jednotlivé podmnožiny. Pri skenovaní ulice je napríklad možné segmentovať mračno na jednotlivé množiny ako je cesta, autá, stromy apod. Pri skenovaní výhradne jedného objektu sa tento krok zvyčajne nevykonáva, avšak pokiaľ daný objekt obsahuje viacero zón, ktoré je účelné rozdeliť, krok segmentácie mračna bodov je možné vykonať aj v tomto prípade.

1.3.6 MODELOVANIE MRAČNA BODOV

Nie vždy je však žiadaným výstupom zo skenovania triangulovaná plocha s množstvom plôch vyjadrujúcich principiálne jednoduchšie tvary. Preto sa niekedy v poslednom kroku pristupuje na kompletne premodelovanie objektu v súlade s daným mračnom bodov pre získanie jednoduchšieho geometrického modelu, ktorý síce nemusí obsahovať toľko detailov avšak zachová základný tvar objektu. Parametre ako sú rozmery objektu sú získané priamo odmeraním zo samotného mračna. Premodelovaním sa však môžu z objektu stratiť rôzne deviácie od ideálneho tvaru a taktiež rôzne chyby ako sú trhliny apod., ktoré môžu byť prítomné v polygonálnej sieti. [5]

2 METÓDA ZÍSKANIA DÁT

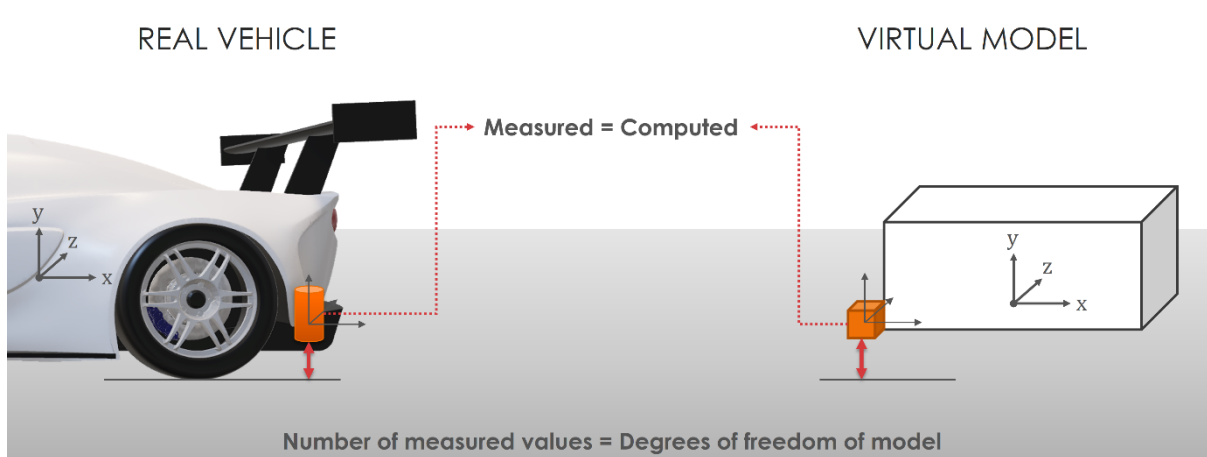
Získanie prvotných dát je realizované niekoľkonásobným prejazdom vozidla osadeného senzormi po vyšetrovanej trati. Modelovanie dát je postavené na troch kľúčových nástrojoch [11]:

Prvým nástrojom je merací systém osadený na vozidle, ktorý zaznamenáva informáciu o všetkých stupňoch voľnosti vozidla, o jeho kinematike a zrýchleniach. Špeciálny merací systém pre tento účel bol vyvinutý na Ústave automobilového a dopravného inžinierstva VUT.

Druhým nástrojom je softvér ktorý umožňuje analýzu získaných dát a taktiež prepojenie s výpočtovým modelom. Túto funkcionality zabezpečuje špeciálny softvér, ktorý je nadstavbou do Matlabu – Telematrix, vyvinutý taktiež na ÚADI VUT.

Tretím nástrojom je softvér pre matematický model vozidla, ktorý musí v záujme presnosti byť čo najpodrobnejší. Pre tento účel je použiteľný softvér ADAMS, ktorý umožňuje vytvoriť veľmi detailné matematické modely vozidiel. Taktiež je však možné použiť aj softvéru SAMS vyvinutého na ÚADI VUT, ktorý je svojou funkcionality veľmi podobný spomínanému softvéru ADAMS. Program je postavený tak, že je možné dáta namerané z vozidla nahráť priamo do výpočtového modelu. Taktiež na rozdiel od ADAMSu umožňuje softvér SAMS aj inverzný režim (výpočet potrebného silového pôsobenia zo znalosti pohybu).

Prepojenie nameraných dát a matematického modelu je postavené na jednoduchej myšlienke – signály zo senzorov prichytených na vozidle sa musia v každom okamžiku pohybu zhodovať so signálmi vypočítanými z matematického modelu, ktorý obsahuje rovnaké senzory – vid' obr. 21. Za tejto podmienky sa vypočíta kompletný dynamický stav, z ktorého je potom možné zistiť mnoho užitočných veličín, ktoré by inak boli veľmi ťažko odmerateľné. [11]



Obr. 21 Prepojenie medzi dátami zo skutočného vozidla a namodelovanými dátami [11]

Výhodou tejto metódy je, že všetky veličiny potrebné na zostavenie modelu pohybu vozidla sú zaznamenávané priamo vozidlom, preto táto metóda nevyžaduje toľko prípravných krokov ktoré sú zvyčajne potrebné pri konvenčnom modelovaní. Taktiež je vhodná pre simulácie jazdy po trati, kde pri konvenčných metódach je model prakticky tak presný ako je namodelovaný, tj. pre čo najlepšie zachytenie skutočnosti musí obsahovať všetky variácie trati. Takéto modelovanie vyžaduje veľké množstvo času. Rovnako je potreba pri takýchto simuláciách uvažovať aj s modelovaním pneumatík a aerodynamiky a taktiež modelu vodiča. Pri

zložitejších reliéfoch tratí ako sú napr. rally trate je takéto modelovanie veľmi zložité. Preto je táto metóda veľmi vhodná aj pre účely profesionálneho pretekania. [11]

3 REKONŠTRUKCIA TRATE

Hlavným cieľom danej práce je rekonštrukcia povrchu trate. Po prvotnej analýze vstupných dát budú pre tento účel v nasledujúcich kapitolách predstavené celkovo tri odlišné prístupy k danej problematike.

3.1 ANALÝZA VSTUPNÝCH DÁT A URČENIE POTREBNÝCH METÓD

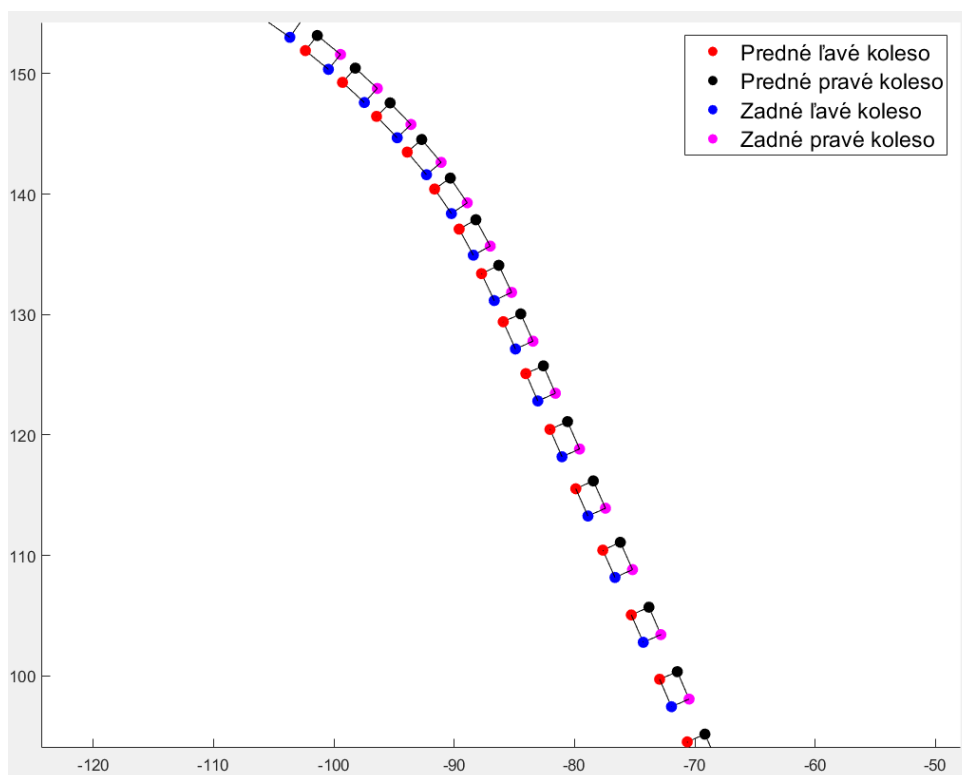
Ešte pred samotným spracovaním je vhodné vykonať analýzu formátu vstupných dát, čím sa môžu detekovať možné nedostatky alebo nepresnosti vo vstupných dátach ktoré ovplyvnia zvolenie metodiky spracovania.

Po analýze vstupných dát, ktoré budú spracované v rámci tohto skriptu je zrejmé, že nasledujú určitý formát. Súbor s dátami je matica s veľkosťou $n \times 12$, kde n je počet jednotlivých zosnímaní, kde každé zosnímanie predstavuje súčasné zapísanie súradníc zo všetkých štyroch kolies vozidla. Prvé tri stĺpce matice zodpovedajú umiestneniu ľavého predného kola na príslušných osiach kartézskoho súradného systému – tj. osiam X, Y a Z. Ďalšie tri trojice stĺpcov následne zodpovedajú ostatným kolesám v poradí pravé predné – ľavé zadné – pravé zadné. Počet riadkov, tj. bodov je prakticky neobmedzený, avšak väčší počet zosnímaných údajov má prirodzene viac či menej citel'ný vplyv na trvanie výpočtu. Väčším počtom vstupných údajov je však všeobecne možné získať presnejší výsledok. Dátový súbor spracovávaný v tejto práci obsahuje 43873 zosnímaní a samotné snímanie prebieha s konštantou frekvenciou 100 Hz.

	1	2	3	4	5	6	7	8	9	10	11	12
1	4.9732	-73.1881	-0.7096	6.3137	-72.3176	-0.7289	6.3457	-75.2583	-0.8259	7.6794	-74.3777	-0.8465
2	4.8868	-73.0146	-0.7029	6.2246	-72.1394	-0.7212	6.2632	-75.0810	-0.8165	7.5950	-74.1975	-0.8375
3	4.8067	-72.8405	-0.6970	6.1420	-71.9611	-0.7141	6.1861	-74.9037	-0.8076	7.5163	-74.0178	-0.8291
4	4.7167	-72.6748	-0.6888	6.0498	-71.7919	-0.7046	6.0980	-74.7355	-0.7964	7.4270	-73.8478	-0.8184
5	4.6302	-72.5015	-0.6836	5.9613	-71.6153	-0.6978	6.0130	-74.5599	-0.7883	7.3411	-73.6709	-0.8106
6	4.5455	-72.3256	-0.6774	5.8749	-71.4366	-0.6903	5.9287	-74.3823	-0.7794	7.2562	-73.4925	-0.8020
7	4.4551	-72.1585	-0.6701	5.7830	-71.2671	-0.6817	5.8382	-74.2138	-0.7696	7.1654	-73.3235	-0.7924
8	4.3637	-71.9836	-0.6643	5.6904	-71.0903	-0.6747	5.7461	-74.0379	-0.7614	7.0733	-73.1476	-0.7842
9	4.2785	-71.8152	-0.6578	5.6045	-70.9207	-0.6673	5.6594	-73.8692	-0.7529	6.9870	-72.9796	-0.7756
10	4.1829	-71.6393	-0.6502	5.5082	-70.7439	-0.6591	5.5620	-73.6932	-0.7435	6.8901	-72.8043	-0.7659

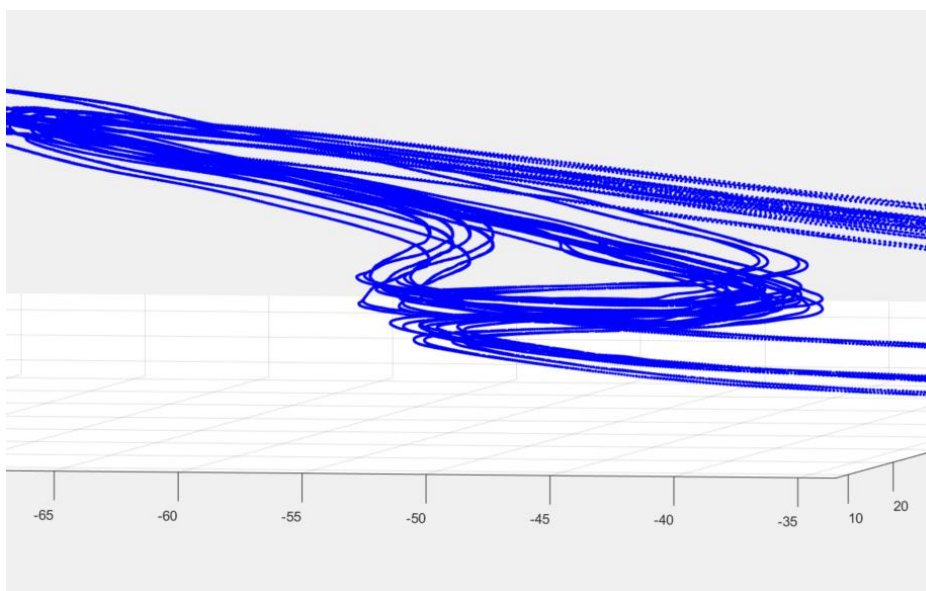
X_{pl} Y_{pl} Z_{pl} X_{pp} Y_{pp} Z_{pp} X_{zl} Y_{zl} Z_{zl} X_{zp} Y_{zp} Z_{zp}
 Predné Ľavé koleso Predné Pravé koleso Zadné Ľavé koleso Zadné Pravé koleso

Obr. 22 Matica vstupných údajov

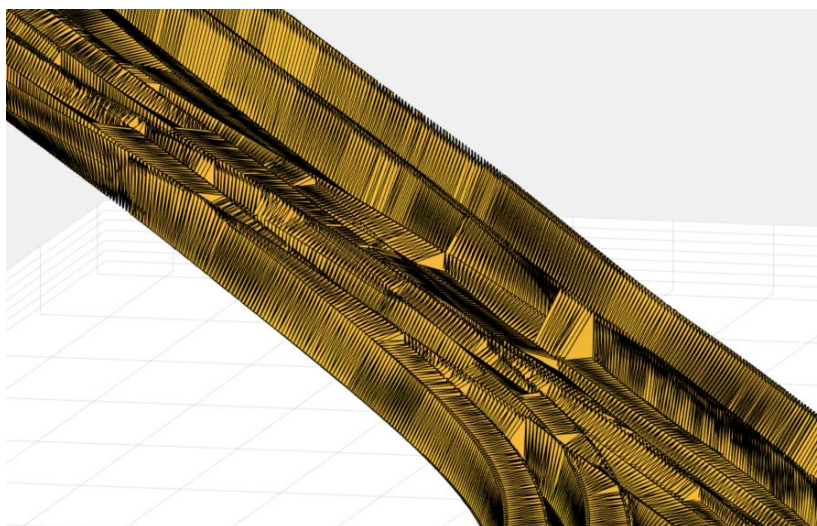


Obr. 23 Vykreslenie časovo preriedenej časti prejazdu

Vykreslenie vstupných dát rovnako poukazuje na pomerne značný obsah šumu, ktorý je prítomný najmä v súradniciach na osi Z – body zosnímané pri rôznych prejazdoch neležia v jednej rovine, ale sú od seba odsadené v Z-smere – viď obr. 24.



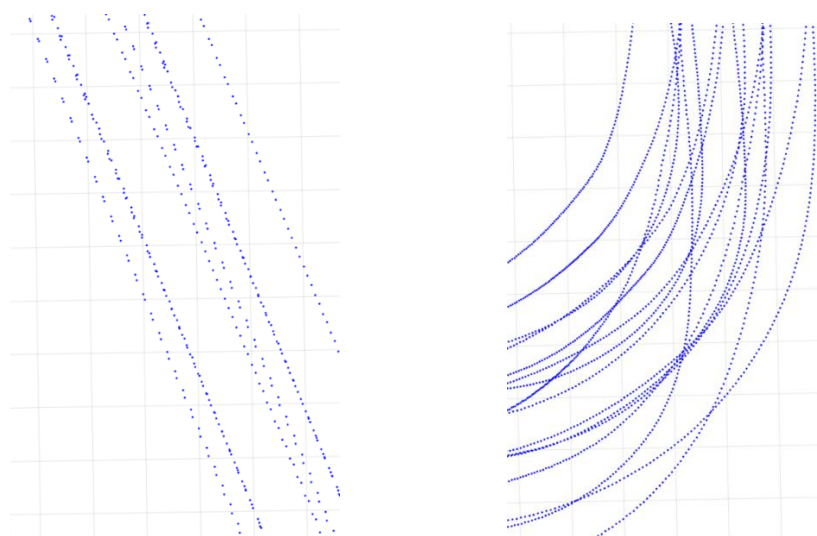
Obr. 24 Prítomný šum v Z-smere



Obr. 25 Triangulácia vstupných dát pre vizualizáciu zašumenia

Tento fakt poukazuje na nutnosť použitia odšumovacieho filtru, ktorý tento šum dokáže zmierniť. Každá z neskôr popísaných metód používa iný algoritmus na odšumovanie a práve algoritmus na odšumovanie je jedným z kľúčových smerodajných faktorov, ktorý líši jednotlivé prístupy.

Ďalším faktorom je objem dát. Ponúknutá matica má veľkosť 43 873 x 12, čo predstavuje 175 492 jednotlivých bodov, čo nepredstavuje príliš výrazné kvantum údajov, avšak objem vstupných dát môže byť aj omnoho väčší, čo môže značne zvýšiť výpočtový čas. Preto je vhodné do skriptu implementovať rovnako aj metódu na redukčné filtrovanie dát (down-sampling) pred samotným procesom rekonštrukcie. Filtrovanie dát má okrem zredukovania výpočtového času rovnako pozitívny vplyv na rovnomernosť súboru bodov – v miestach veľkého zahustenia sú body preriedené. Nakoľko vozidlo sníma stále s rovnakou frekvenciou, ktorá nezávisí na rýchlosti auta, je zrejmé, že hustota bodov v zákrutách, kde sa vozidlo pohybovalo pomalšie je väčšia ako na rovných častiach trate kde sa vozidlo pohybovalo rýchlejšie. Taktiež, ak vozidlo prechádzalo niektorými časťami trate viac ako inými, aj toto má za následok hustotnú nekonzistenciu súboru bodov.



Obr. 26 Hustota bodov pri rovnakom priblížení na rovnej časti trate a v zákrute

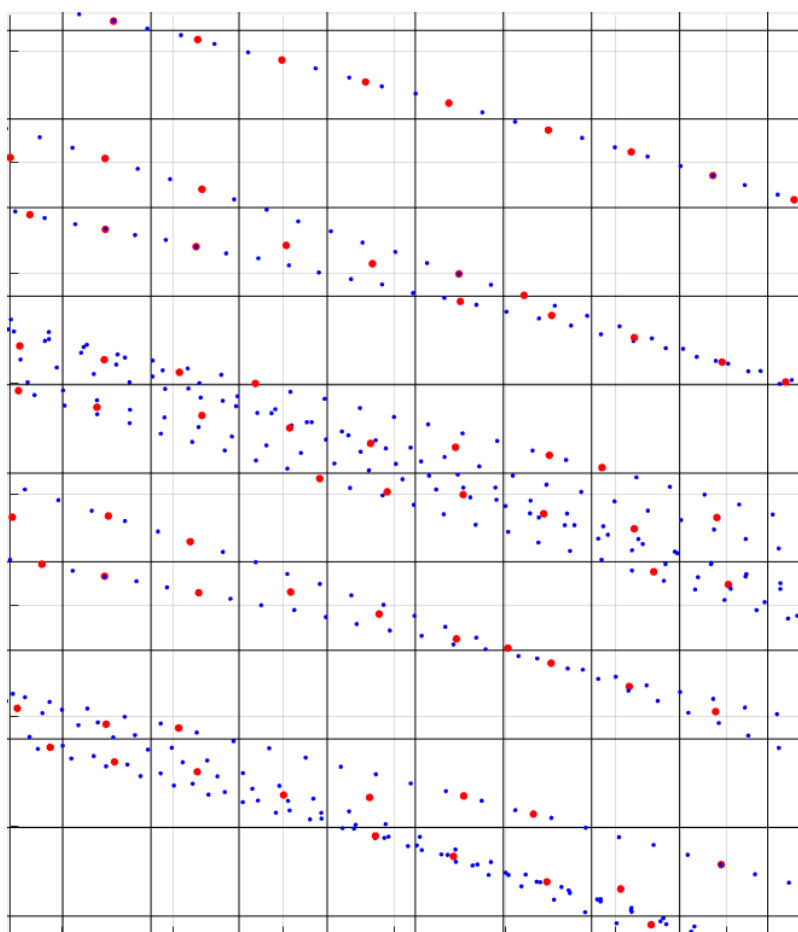
3.2 PRÍSTUP 1 – REKONŠTRUKCIA POMOCOU BILATERÁLNEHO FILTRA

Prvý prístup spočíva v aplikácii bilaterálneho filtra, ktorý je vhodný na odšumovanie dát a je priamo aplikovateľný na mračno bodov. Bilaterálny filter ako vstup používa informácie o susedných bodoch každého bodu a o smeroch normál v každom bode mračna, ktoré je potreba prvotne získať.

3.2.1 FILTRÁCIA MRAČNA

Redukčná filtrácia bodov mračna má za úlohu zredukovať množstvo dát pri zachovaní čo najväčšej miery pôvodnej informácie. Hlavným efektom je zníženie výpočtového času. Tento krok nie je nutný, avšak práve pri tomto prístupe je veľmi vhodný, nakoľko ďalším efektom filtrácie je, ako bolo spomenuté v časti 1.3.3 zrovnomenenie hustoty bodov, čo je výhodné ako z hľadiska finálnej konzistentnosti triangulovanej plochy, tak aj z hľadiska princípu funkcie daného filtra.

Filtrácia sa uskutočňuje pomocou zaradenia bodov do voxelovej siete (vid' kapitola 1.3.3) so zvolenou jemnosťou a nahradenia všetkých bodov v každom voxeli jednotlivými bodmi, ktoré sa získajú priemerom všetkých bodov – substitučný bod predstavuje ťažisko všetkých bodov v danom voxeli.



Obr. 27 Nahradenie bodov v každom voxeli jednotlivými ťažiskami. Modré – pôvodné body, červené – substitučné body po prefiltrovaní

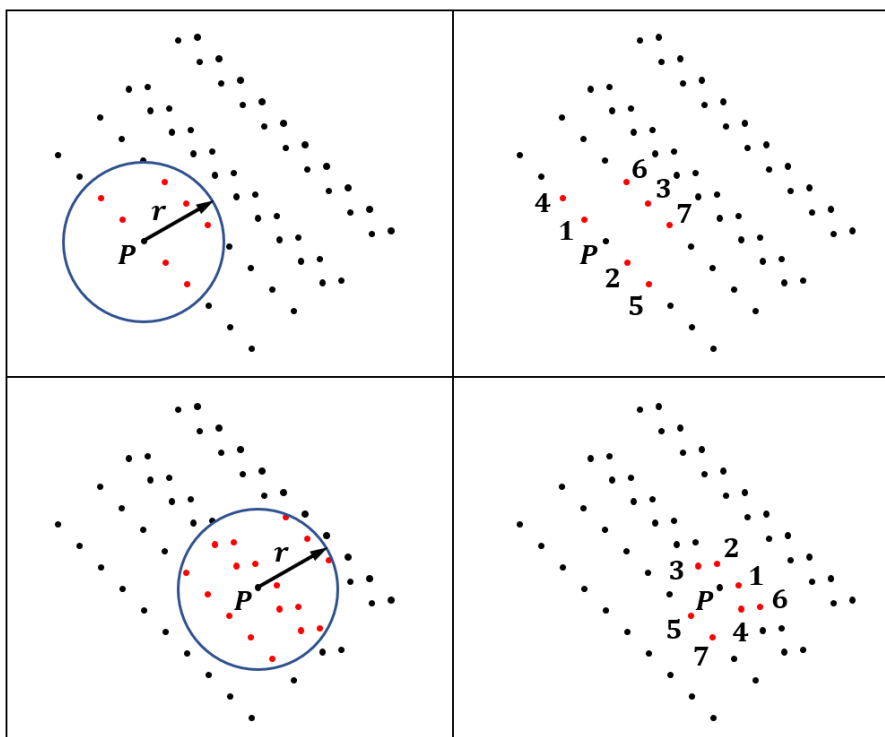
3.2.2 ZÍSKANIE SUSEDNÝCH BODOV KU KAŽDÉMU BODU

Ďalším krokom je nájdenie susedných bodov pre každý bod v súbore bodov.

Prvým logickým riešením tejto problematiky, ktoré sa ponúka je tzv. naivný prístup, pri ktorom sa vypočítajú vzdialenosti každého bodu mračna ku všetkým ostatným bodom a následne sa zoradia podľa vzdialenosti. Z takejto matice je potom možný výber potrebného počtu najbližších bodov ku každému bodu. Táto metóda však získala svoj názov najmä z toho dôvodu, že takýto prístup je veľmi neefektívny a oproti ostatným metódam vyžaduje rádovo vyššie výpočtové časy. Tento prístup je vhodný pre malé súbory bodov, kedy výpočtový čas takéhoto jednoduchého algoritmu bude nižší ako čas potrebný na zostavenie sofistikovanejších metód.

Pre väčšie súbory je preto nutné použiť alternatívne metódy. Z hľadiska hľadania najbližších bodov k danému bodu je možné sa na problematiku pozrieť dvojako - hľadáme buď:

- užívateľom určený počet „ k “ najbližších bodov ku každému bodu, pričom ich skutočná absolútna vzdialenosť od skúmaného bodu nie je podstatná alebo
- všetky body v užívateľom zvolenom okolí bodu - v 3D priestore sa jedná o guľu s priemerom „ r “, pričom ich počet nie je podstatný.



Obr. 28 Rozdiel rNN a kNN na rovnakom súbore bodov pre rovnaký bod P . Vľavo – metóda rNN, vpravo – metóda kNN ($k = 7$). Hore – bod P na kraji (menšia hustota), dole – bod P je v strede (väčšia hustota)

Oba prístupy v konečnom dôsledku, rovnako ako naivný prístup, počítajú pre každý bod vzdialenosti k iným bodom, avšak podľa stanovenia parametrov je tento výber skúmaných bodov výrazne redukovaný pomocou obmedzenia na vyšetrovanie iba množiny potenciálnych najbližších susedov. To sa väčšinou realizuje na princípe subdivízie súboru bodov na viacero častí a následne prehľadanie iba tých častí, ktoré môžu potenciálne obsahovať najbližšie body.

Obidve metódy sú vypracované v tejto práci a radia sa medzi presné metódy (tj. nájdú vždy skutočne najbližšie body, na rozdiel od aproximačných metód ktoré pracujú s určitou podmienenou nepresnosťou a za cenu nižších časových nárokov môžu prehliadnúť niektoré najbližšie body a zaradiť medzi najbližšie body body, ktoré tam v skutočnosti nepatria).

METÓDA KNN

Prvým použitým prístupom je metóda kNN (z angl. k-Nearest Neighbours – „k najbližších susedov“), ktorá pre každý bod nájde užívateľom konkrétne definovaný počet najbližších susedných bodov. Tento počet je hlavným a jediným vstupom do algoritmu kNN, okrem vyšetřovaného súboru bodov. Výstupom z tohto algoritmu je matica s veľkosťou:

$$NN = [n \times k], \quad (6)$$

kde n – počet všetkých vyšetřovaných bodov, k – definovaný počet požadovaných najbližších susedných bodov

Hodnoty matice sú indexy bodov z matice XYZ, ktoré patria do množiny najbližších bodov k danému bodu, tj. na mieste $NN(i,j)$ sa vyskytuje index j -tého najbližšieho bodu k i -tému vyšetřovanému bodu.

Pri hľadaní k najbližších bodov nie je braný ohľad na absolútnu vzdialenosť susedných bodov k vyšetřovanému bodu, iba na ich celkový počet.

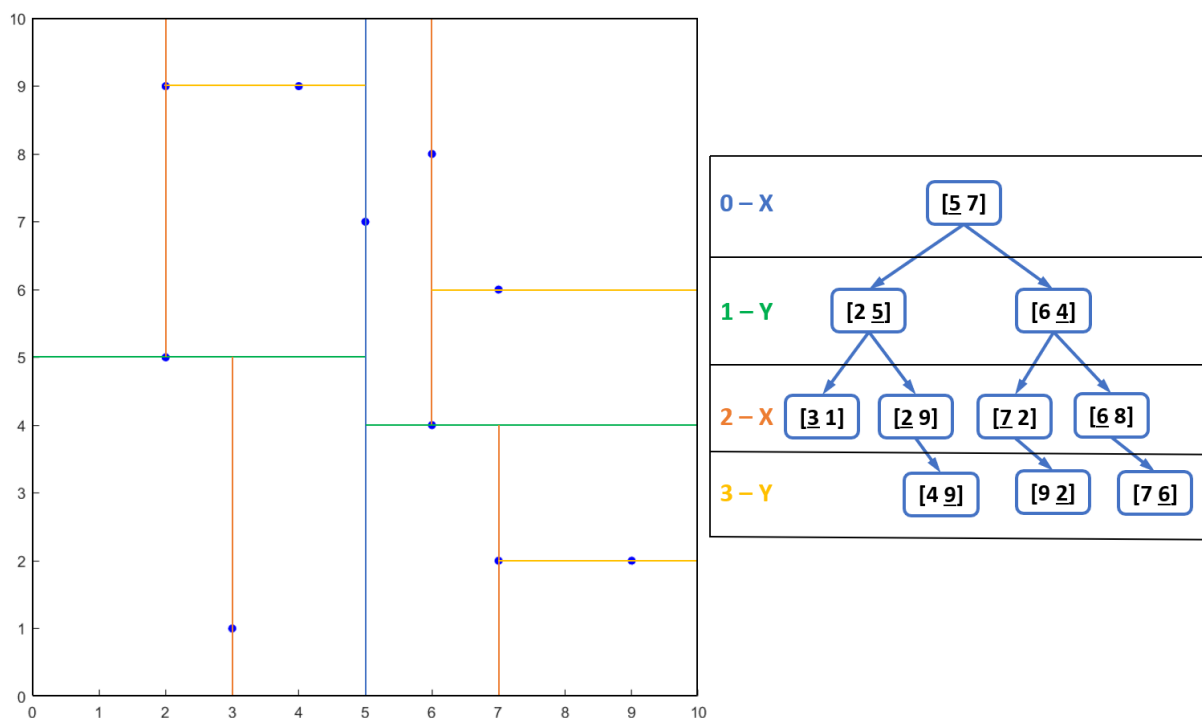
Ako bolo vyššie spomenuté, dôležitú úlohu zohráva správne priestorové rozdelenie mračna bodov a tiež správne prehľadanie všetkých potrebných jednotlivých množín bodov, ktoré môžu obsahovať najbližší bod. Jedným vhodným nástrojom, ktorý je možné využiť je KD-strom.

KD strom (z angl. K-Dimensional tree – K-rozmerný strom – v tomto prípade 3-rozmerný strom) je nástroj na rozdelenie súboru bodov na presne ohraňované oblasti, pričom samotný proces delenia prebieha špecifickým spôsobom. Pre jednoduchosť bude ilustrovaný na príklade v rovine (tj. vytvorenie 2-rozmerného stromu), avšak analogicky je možné použiť rovnaký postup aj pre trojrozmerný súbor dát. Delenie do KD stromu prebieha v krokoch a v každom kroku je súbor bodov obsiahnutý v danej oblasti rozdelený na polovicu, pričom deliaca rovina prechádza cez mediánový bod aby sa dosiahlo toho, že na každej strane delenia ostane rovnaký počet bodov. Pokiaľ je v danom kroku párny počet bodov, tj. mediánom je hodnota medzi dvoma strednými bodmi, bod s nižšou hodnotou je zvolený ako deliaci bod – deliaca rovina musí vždy prechádzať bodom. Tým sa minimalizuje počet deliacich úrovní.

Delenie do KD stromu začína výberom počiatkovej osi a spôsobom nasledovania delenia – konvenčne sa začína delením podľa súradníc na osi X, nasledovaným delením podľa osí Y a Z. V prvom kroku, nakoľko začína delením podľa súradníc na osi X, sa zoradia všetky body ktoré obsahuje daná oblasť (na začiatku celý súbor dát) podľa súradnice X a zoradia sa podľa tejto súradnice. Bod, ktorý sa nachádza v strede (medián) je kľúčový nakoľko práve tento bod určuje polohu deliacej roviny. Rozdelením jedného počiatkového súboru dát na dva súbory s rovnakým počtom bodov je prvá úroveň delenia ukončená. Nasleduje druhý krok, kedy sa zostavuje druhá úroveň stromu. Nakoľko sa v prvej úrovni delilo pomocou súradnice X, tentokrát sa bude deliť podľa súradnice Y. Postup je totožný, avšak tentoraz sa delia už dve oblasti, z ktorých je nutné každú rozdeliť podľa ich vlastného mediánového bodu vzhľadom na súradnicu Y. Tým sa získajú celkovo 4 menšie oblasti. Takto sa postupuje do ďalšej úrovne, kedy sú znova delené všetky novovzniknuté oblasti, tentokrát už podľa súradnice Z. Akonáhle

sa vykoná delenie podľa všetkých súradníc, algoritmus delenia sa vracia znova na delenie podľa súradnice X. Takto proces cykluje počas celého procesu delenia až kým sa delenie nezastaví. Delenie sa zastaví, pokiaľ v danej oblasti už nezostanú žiadne body na delenie.

Tým, že pri každom procese delenia sa deliaca rovina prekladá bodom, tieto deliace body už nie sú ďalej pri delení uvažované, nakoľko „nepatria“ ani do jednej z oblastí. Tým sa počet vyšetřovaných bodov v každom kroku znižuje.



Obr. 29 Rozdelenie dvojrozmernej množiny bodov pomocou KD-stromu

Na obr. 29 je znázornené rozdelenie dvojrozmerného súboru dát do KD-stromu. Prvou deliacou súradnicou je X, pričom v ilustrovanom prípade sú v prípade párneho počtu bodov vyberané ako deliace body body s menšou hodnotou na danej súradnici, čo je zreteľné na prvom deliacom bode so súradnicami [5 7] a taktiež na deliacom bode v druhej úrovni so súradnicami [2 5]. Delenie skončilo po 3 úrovniach, kedy sa v žiadnej z novovzniknutých oblastí už nevyskytoval ani jeden bod.

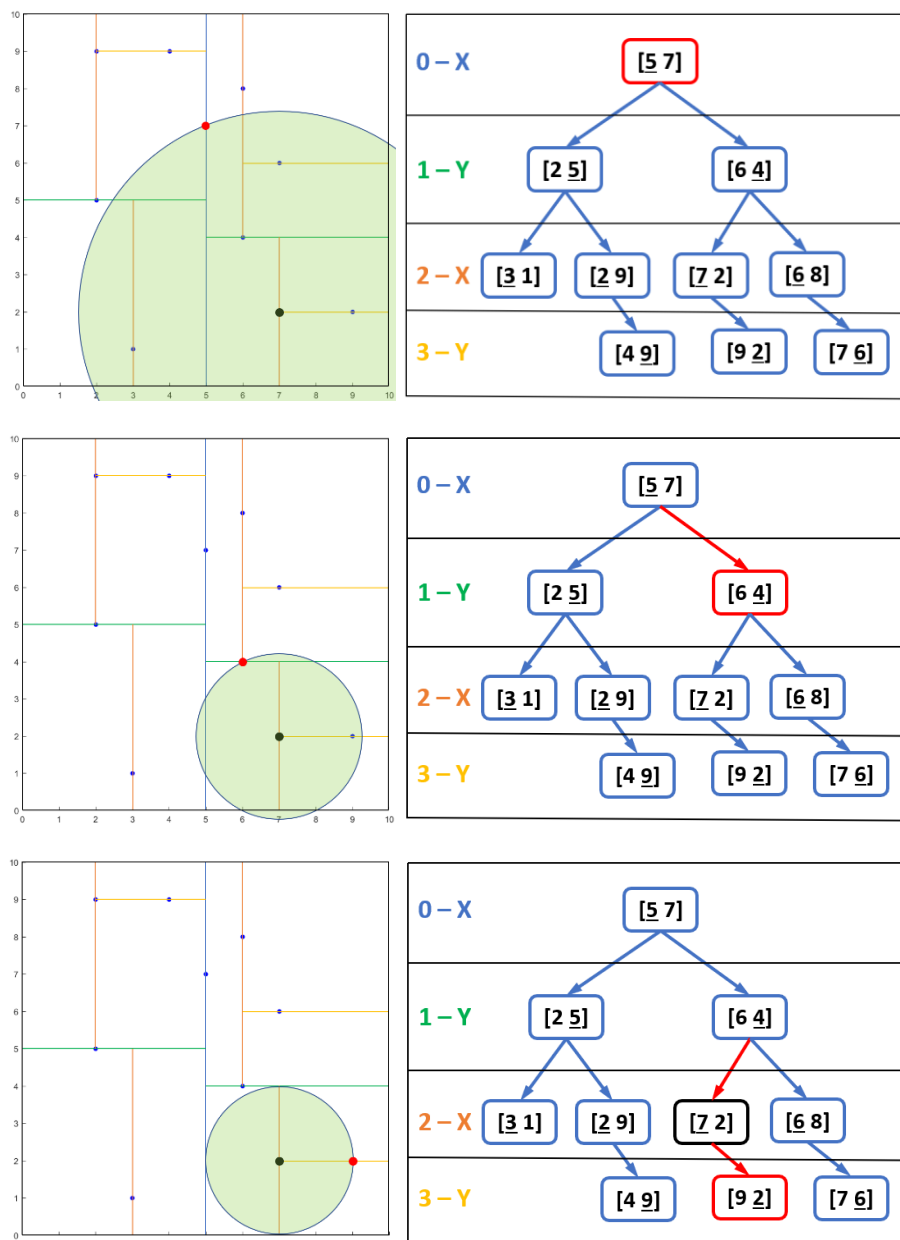
Otázka však nastáva, ako je možné takéto rozdelenie efektívne využiť na hľadanie najbližších susedných bodov. Ukazuje sa, že práve KD-strom je vhodným nástrojom na hľadanie susedných bodov. [12] Nasleduje predom definovaný logický postup. Povedzme že je potreba nájsť „k“ susedných bodov k vyšetřovanému bodu „i“. Potom algoritmus postupuje takto:

1. Vytvorí sa vektor s veľkosťou $1 \times k$, do ktorého budú zaznamenávané doterajšie najbližšie nájdené body, pričom body budú zoradené vzostupne podľa ich vzdialenosti k vyšetřovanému bodu – bod s indexom „1“ je najbližšie a bod s indexom „k“ je najďalej. Tento vektor sa bude v priebehu hľadania bodov vplyvom nachádzania nových bodov neustále meniť až do tej miery, kedy bude istota, že v ostatných neprehľadaných oblastiach už nemôže byť bod, ktorý môže byť k – najbližším susedným bodom.
2. Algoritmus postupuje v strome smerom od koreňového deliaceho bodu (na najvyššej,

prvej úrovni) smerom k bodu, ktorý je aktuálne vyšetrovaný, pričom v každom uzle sa počíta vzdialenosť medzi daným uzlovým bodom a vyšetrovaným bodom. Ak je táto vzdialenosť menšia ako vzdialenosť k doterajšiemu najvzdialenejšiemu zaznamenanému susednému bodu (k -tý – posledný bod vo vektore najbližších bodov), daný bod sa zapíše do vektora nájdených bodov a vektor sa znova zoradí. Vetva, po ktorej algoritmus v každom uzle pokračuje sa určí jednoduchým porovnaním deliacej súradnice tohto uzlového bodu a odpovedajúcej súradnice vyšetrovaného bodu – nakoľko každý uzol (bod) predstavuje jednu deliacu súradnicu. V praxi je možné automaticky zaznamenať prvých k – vyšetrených bodov aj bez ich porovnávania, nakoľko vektor najbližších bodov na začiatku ešte neobsahuje žiadne najbližšie body a týchto prvých k bodov budú automaticky predbežne najlepšie bez ohľadu na ich vzdialenosti (ktoré je však napriek tomu nutné spočítať pre neskoršie porovnávanie).

3. Keď sa algoritmus dostane po vyšetrovaný bod, pokračuje v strome nižšie na vopred zvolenú stranu (v tomto prípade môže byť ľubovoľná, avšak je vhodné aby bola konzistentná) – samozrejme za predpokladu ak je daný bod uzlovým bodom a nie listovým – v tom prípade sa tento krok preskakuje a pokračuje sa ďalej krokom 4.
4. Keď sa algoritmus dostane po listový bod, postupuje v strome naspäť – smerom nahor, pričom v každom uzle sa skontroluje, či vzdialenosť vyšetrovaného bodu k deliacej rovine daného uzlového bodu je menšia ako vzdialenosť k najvzdialenejšiemu bodu z množiny nájdených k – najbližších bodov. Pokiaľ je táto vzdialenosť menšia, algoritmus pokračuje smerom dole do tejto vetvy nakoľko môže potenciálne obsahovať bod, ktorý patrí do množiny k – najbližších bodov. Pokiaľ je táto deliaca rovina ďalej, v tom prípade je zbytočné túto vetvu (a všetky podradené body) vyšetrovať, nakoľko v tej oblasti nemôže byť bližší bod ako tie ktoré boli doteraz nájdené.
5. Pokiaľ v akomkoľvek uzlovom bode algoritmus pokračuje do druhej, nevyšetrenej vetvy, pokračuje sa v súlade s krokom 3. Pokiaľ danú vetvu nie je dôvod vyšetrovať, pokračuje sa zase do vyššieho uzlu a znova sa skontroluje podmienka popísaná v kroku 4.
6. Hľadanie NN končí pokiaľ všetky vetvy potenciálne obsahujúce NN boli prehľadané.

Príklad: Hľadá sa najbližší susedný bod pre bod $[7 \ 2]$ zo súboru bodov na obr. 29. Po každom kroku je červeným označený doterajší najbližší nájdený bod. Kruh zaznačuje oblasť, v ktorej sa potenciálne môžu nachádzať ešte bližšie body, tj. ak kruh zasahuje časťou do nejakej oblasti, je ju nutné preskúmať. Po každom nájdení bližšieho bodu sa táto oblasť zmenšuje a zmenšuje sa aj množina oblastí, ktoré musia byť prehľadané. Pre tento prípad hľadáme síce iba jeden najbližší susedný bod, avšak jedinou zmenou pri „ k “ najbližších bodoch je, že je polomer tohto kruhu daný najvzdialenejším nájdeným susedným bodom (tj. k -tým), inak je postup identický.



Obr. 30 Vizualizácia hľadania v KD-strome

METÓDA RNN

Alternatívnym prístupom k hľadaniu najbližších susedných bodov je použitie metódy hľadania všetkých bodov v určitom rádiuse. Tento, užívateľom zadefinovaný polomer „ r “ je jediným vstupom do tohto algoritmu, okrem vyšetrovaného súboru bodov. Výstupom z tohto algoritmu je teda matica s veľkosťou:

$$NN = [n \times n_r], \quad (7)$$

kde n – počet všetkých vyšetrovaných bodov

n_r – najväčší počet susedných bodov pre vyšetrované body

Hodnoty matice sú indexy bodov z matice XYZ, ktoré patria do množiny najbližších bodov

k danému bodu, tj. na mieste $NN(i,j)$ sa vyskytuje index j-tého najbližšieho bodu k i-tému vyšetrovanému bodu.

Je nutné poznamenať, že body môžu mať rozličný počet najbližších susedných bodov, čo môže byť ovplyvnené napr. hustotou siete alebo polohou bodu, ako je to ukázané na obr. 28. Preto počet stĺpcov matice nie je tak jednoduché predpovedať.

Rovnako ako u metódy kNN, aj táto metóda sa opiera o využitie vhodnej metódy rozdelenia priestoru. V tomto prípade bude využité rozdelenie priestoru pomocou voxelovej siete.

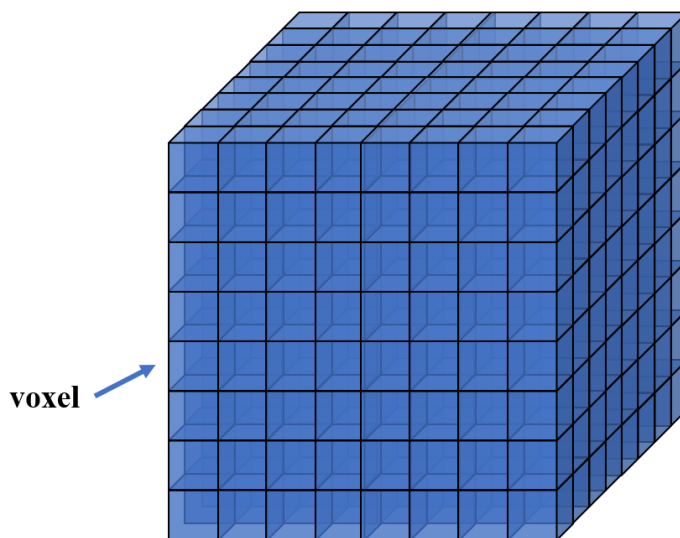
Pre každý súbor bodov je možné určiť jeho ohraničujúci priestor, tj. najmenší možný pomyselný hranol, ktorý obklopuje všetky body súboru bodov. Jeho hranice v troch osiach sú dané jednoducho:

$$X = [\min(X) \max(X)], \quad (8)$$

$$Y = [\min(Y) \max(Y)], \quad (9)$$

$$Z = [\min(Z) \max(Z)]. \quad (10)$$

Pokiaľ sa táto oblasť rozdelí na určený počet rovnako veľkých dielikov, získame rovnomernú voxelovú sieť. Počet dielikov, rovnako ako ich veľkosť na každej z osí nemusí byť rovnaký. Je ale potrebné aby boli všetky dieliky rovnako veľké.



Obr. 31 Voxelová sieť ktorej dieliky sú pravidelné kocky

Práve takáto voxelová sieť bude slúžiť na zaradenie a hľadanie bodov. Zostavením voxelovej siete nadobudne každý voxel tejto siete svoj index. Na základe tohto indexu a znalosti spôsobu zostavovania voxelovej siete sú známe okraje oblasti, ktorú daný voxel ohraničuje. Po jej zostavení je druhým krokom priradenie všetkých bodov zo súboru bodov k voxelom, do ktorých patria a v ďalšom kroku potom podľa znalosti, ktoré body sa nachádzajú v ktorom voxelu sú vyhladané najbližšie susedné body.

Vytvorenie samotnej voxelovej siete sa riadi v tomto prípade jediným faktorom – užívateľom nastaviteľný parameter „r“. Totiž veľkosť dielikov musí byť nie menšia ako rádius „r“ aby sa

zabezpečilo, že všetky susedné body budú vždy nájdené, vid' obr. 32

Priradenie bodov prebieha podľa jednoduchého vzorca, ktorý berie do úvahy spôsob indexovania voxelov pre určenie polohy:

$$PointinVoxel = (x - 1) * n_y * n_z + (y - 1) * n_z + z, \quad (11)$$

kde:

$$x = \frac{(X - \min(X))}{d_x}, \quad (12)$$

$$y = \frac{(Y - \min(Y))}{d_y}, \quad (13)$$

$$z = \frac{(Z - \min(Z))}{d_z}, \quad (14)$$

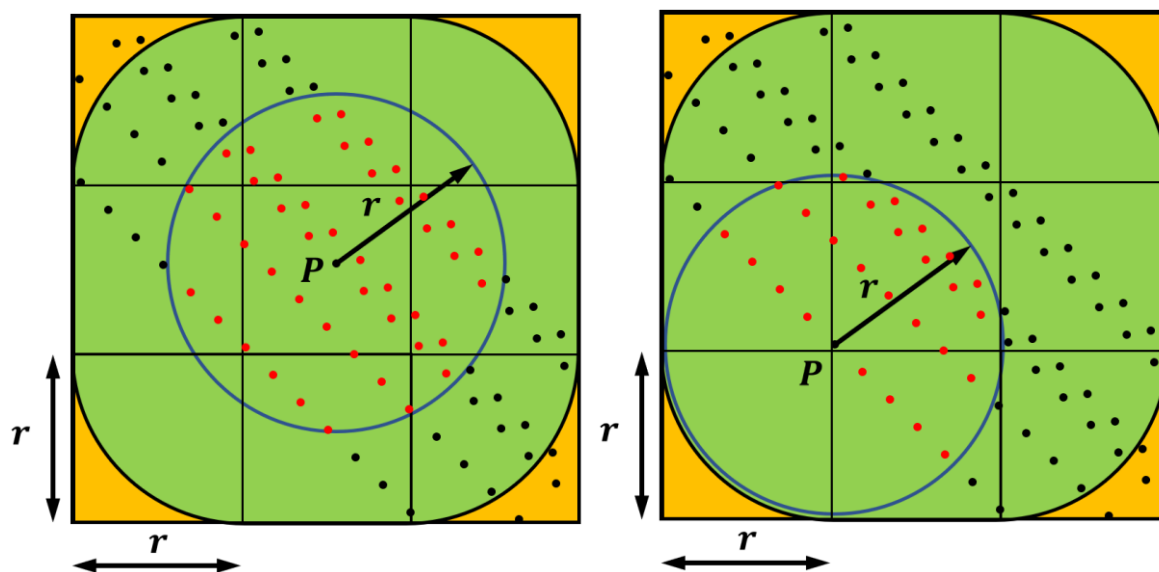
kde: X, Y, Z – vektory súradníc jednotlivých bodov mračna,

n_x, n_y, n_z – počet dielikov na každej z osí,

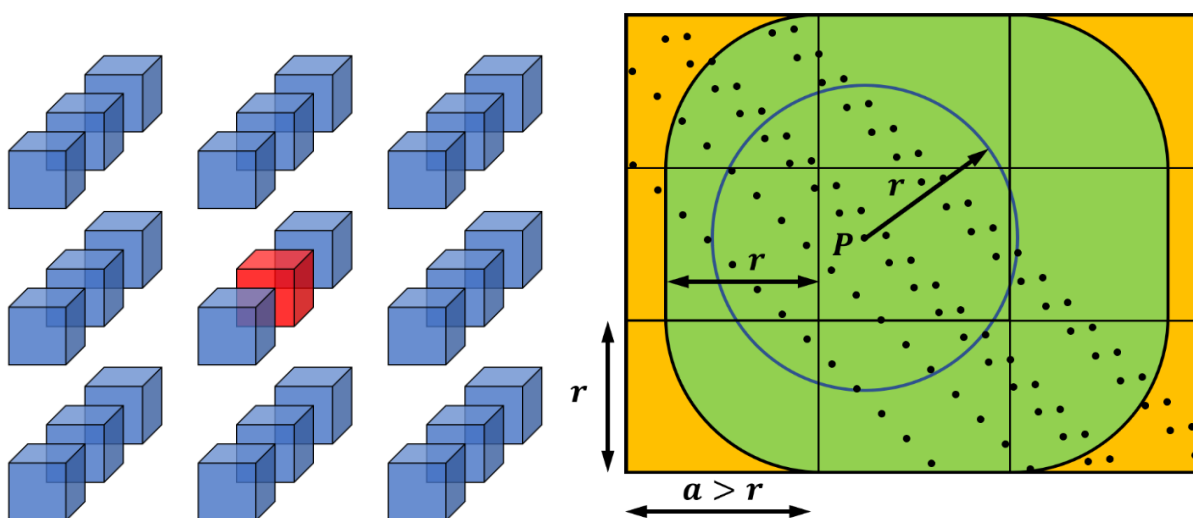
d_x, d_y, d_z – veľkosť dielikov na každej z osí.

Keď sú všetky body priradené k voxelom, nasleduje proces hľadania NN. Algoritmus jednoducho pre každý bod vypočíta vzdialenosti medzi týmto bodom a všetkými bodmi obsiahnutými vo voxelu, v ktorom sa daný bod nachádza a zároveň vo všetkých jeho 26 susedných voxeloch, vid' obr. 33. Pokiaľ je vzdialenosť ku konkrétnemu bodu menšia alebo rovná ako parameter „r“, bod sa zapíše do vektoru najbližších bodov. Tým, že veľkosť strán voxelov je rovná polomeru „r“, tak prehl'adaním aktuálneho voxelu a jeho susedných sa zaručí, že sa prehl'adajú všetky potenciálne body vo vzdialenosti „r“, aj v prípade že by bod ležal na kraji voxelu. Najväčšej efektivity algoritmu sa dosiahne pokiaľ sú voxely čo najbližšie podobné forme kocky, nakoľko pri väčšej nerovnomernosti dĺžky strán voxelu sa môže stať, že bude prehl'adané zbytočne veľké množstvo bodov, ktoré sú vzdialené od vyšetřovaného bodu omnoho ďalej ako parameter r (vid' obr. 34). Preto je v danom algoritme definované:

$$d_x = d_y = d_z = r \quad (15)$$



Obr. 32 Hľadanie susedných bodov pomocou rNN. Vľavo – bod P je v strede voxelu. Vpravo – bod P je na kraji voxelu. Zelená – zóna ktorá môže obsahovať najbližšie body pre body stredného voxelu. Oranžová – zóna ktorá nemôže obsahovať najbližšie body pre body stredného voxelu.



Obr. 33 Voxel a jeho 26 príslušných susedov

Obr. 34 Nerovnomerná veľkosť voxelov a dopad na prehľadane zóny

3.2.3 URČENIE NORMÁL

Druhým vstupom pre algoritmus bilaterálneho filtra je smer normál v každom bode. Tie sa určia aproximáciou práve pomocou zistených najbližších susedných bodov. Aproximácia smeru normálového vektora v akomkoľvek bode zo súboru bodov je realizovaná preložením regresnej plochy pomocou daného bodu a jeho susedných bodov s cieľom minimalizovania sumy (normálových) kvadratických vzdialeností bodov od tejto plochy.

3.2.4 BILATERÁLNE FILTROVANIE

V momente pokiaľ sú známe susedné body a normálové vektory každého bodu prichádza na

rad samotný proces bilaterálneho odšumovania. Princípom bilaterálneho filtrovania je, jednoducho povedané, posunutie každého bodu súboru bodov v smere jeho normálového vektora, kde orientáciu a magnitúdu posunutia určujú práve jeho susedné body, pričom každý susedný bod má inú váhu, tj. mieru vplyvu na výsledné posunutie vyšetrovaného bodu, vid' kap. 1.3.2.

3.2.5 TRIANGULÁCIA DÁT

Posledným krokom spracovania dát je ich triangulácia pre získanie vyjadrenia reliéfu vozovky. V podstate ide o získanie matice:

$$ConnectivityList = [ne \times 2], \quad (16)$$

kde ne – počet jednotlivých hrán triangulovanej plochy – v každom riadku nájdeme indexy dvoch bodov, ktoré majú byť trianguláciou prepojené spoločnou hranou.

V tomto prípade je triangulácia uskutočnená pomocou vstavanej funkcie Matlabu na trianguláciu metódou Alpha shapes, vid' kapitola 1.3.4.

3.3 PRÍSTUP 2 – REKONŠTRUKCIA POMOCOU LAPLACOVEHO ODŠUMOVANIA

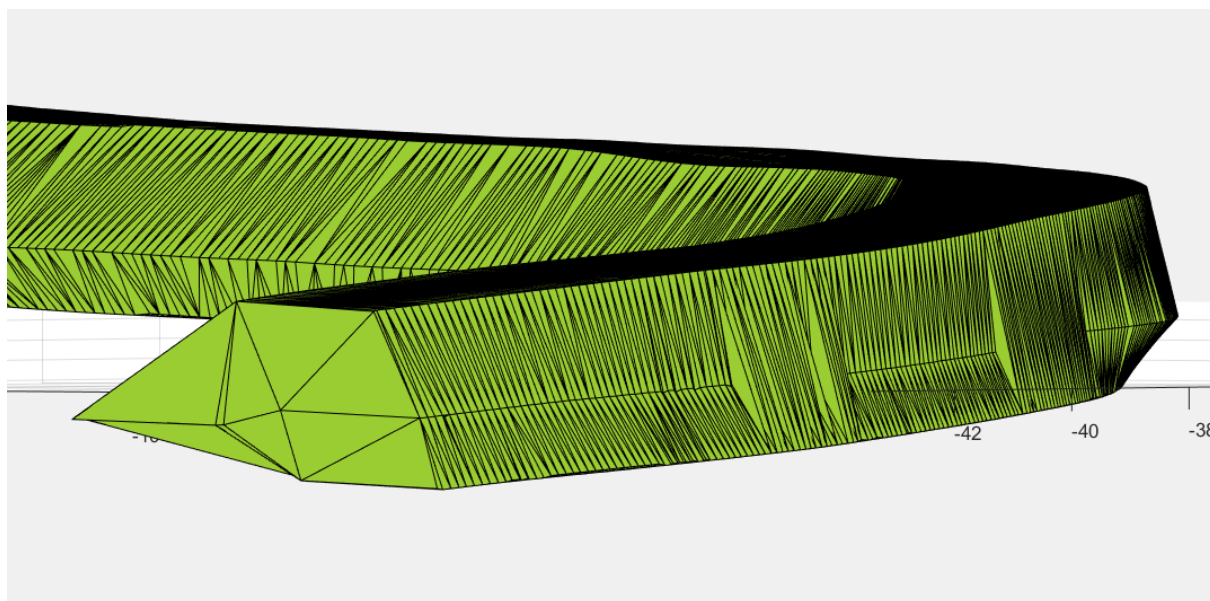
Druhým prístupom pre odšumenie súboru bodov je použitie Laplacoveho odšumovania. Táto metóda je hojne používaná pri odšumovaní sietí pre jej jednoduchosť a účinnosť. [8] Na rozdiel od prvej metódy sa proces odšumovania neuskutočňuje priamo na súbore bodov, ale na už triangulovanej ploche, tzn. že topológia siete sa určí hneď na začiatku. Vstupmi do samotného procesu Laplacoveho odšumovania sú okrem odšumovacích parametrov a počtu iterácií odšumovania vopred triangulovaná plocha zo súboru dát a susedné body prvého rádu pre každý z bodov.

3.3.1 FILTRÁCIA MRAČNA

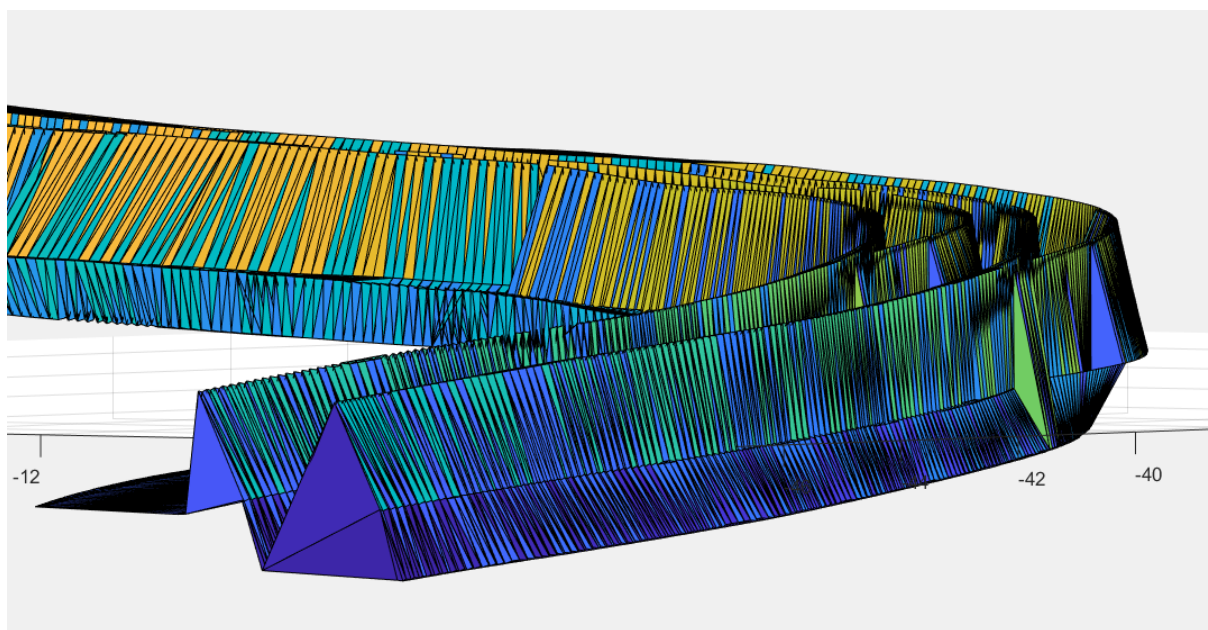
Ako aj u prvej metódy, aj tu je možné na začiatku využiť filtráciu bodov mračna majúce za úlohu zredukovať množstvo dát pri zachovaní čo najväčšej miery pôvodnej informácie, za účelom zníženia výpočtového času a zrovnomenenia hustoty bodov. Tento krok však nie je nutný. Pre princíp začiatočného filtrovania mračna vid' kapitola 1.3.3.

3.3.2 TRIANGULÁCIA

Zatiaľ čo predchádzajúca metóda trianguláciou končila, metóda Laplacoveho odšumovania ňou začína. Tu je však treba väčšej obozretnosti pri triangulácii, nakoľko triangulácia prebieha na neodšumenom dátovom súbore, ktorý môže obsahovať veľkú úroveň šumu a tým výrazne ovplyvniť proces hľadania vhodných spojných hrán. Môže sa napríklad stať, pokiaľ dáta obsahujú veľké výkyvy v Z-smere, že sa v niektorých miestach bude nad sebou objavovať viacero plôch (vid' obr 35). Avšak, nakoľko je zrejmé z charakteru snímaných dát, že na jednom mieste môže byť v Z-smere iba jedna plocha, je vhodnejšie trianguláciu vykonať iba v XY rovine, čím sa zabezpečí, že body budú triangulované bez ohľadu na súradnice na osi Z. Podobne ako v prípade prvej metódy, aj tu bola zvolená triangulácia pomocou Alpha shapes.



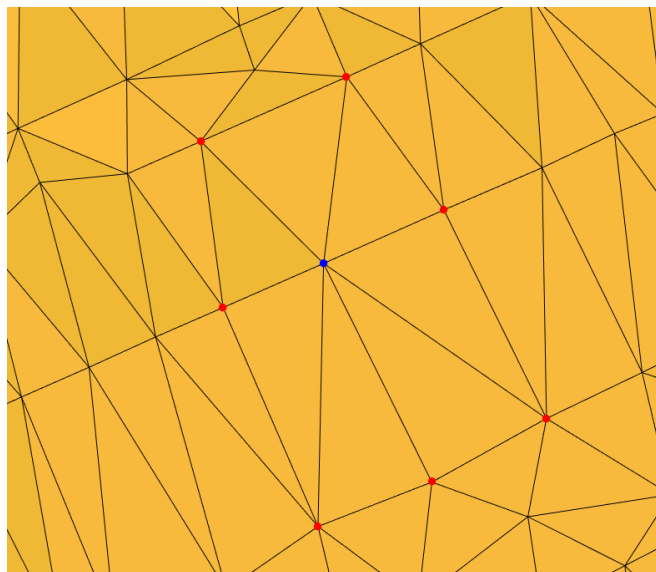
Obr. 35 Nesprávna triangulácia – v Z-smere sa vyskytuje viacero plôch



Obr. 36 Správna triangulácia – vždy iba jeden polygón v Z-smere

3.3.3 NÁJDENIE SUSEDNÝCH BODOV PRVÉHO RÁDU

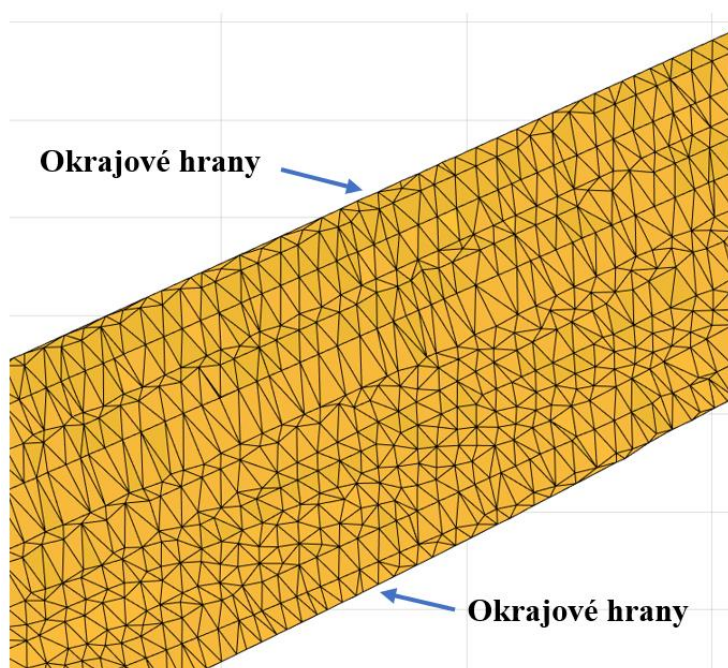
Po vykonaní triangulácie je možné nájsť pre každý vrchol (bod) jeho susedné vrcholy prvého rádu – body, ktoré s vyšetrovaným bodom zdieľajú hranu triangulácie (viď obr. 37). Tu je možné jednoducho využiť trianguláciou vytvorenú maticu ConnectivityList – matica, ktorá obsahuje všetky hrany triangulácie – dvojice indexov bodov ktoré tieto hrany tvoria. Stačí nájsť všetky riadky, kde sa vyskytuje daný index bodu a zapísať indexy bodov, pri ktorých sa v daných riadkoch nachádza.



Obr. 37 Červené body sú susednými vrcholmi prvého rádu pre modrý bod

3.3.4 NÁJDENIE OKRAJOVÝCH HRÁN TRIANGULÁCIE

Z hľadiska funkcie algoritmu Laplacoveho odšumovania je taktiež výhodné nájsť okrajové hrany triangulácie, tj. hrany ležiace na okraji triangulovanej siete. Každá hrana patriaca iba jednej stene je hrana okrajová (viď obr. 38).



Obr. 38 Okrajové hrany triangulácie

3.3.5 LAPLACOVO ODŠUMOVANIE

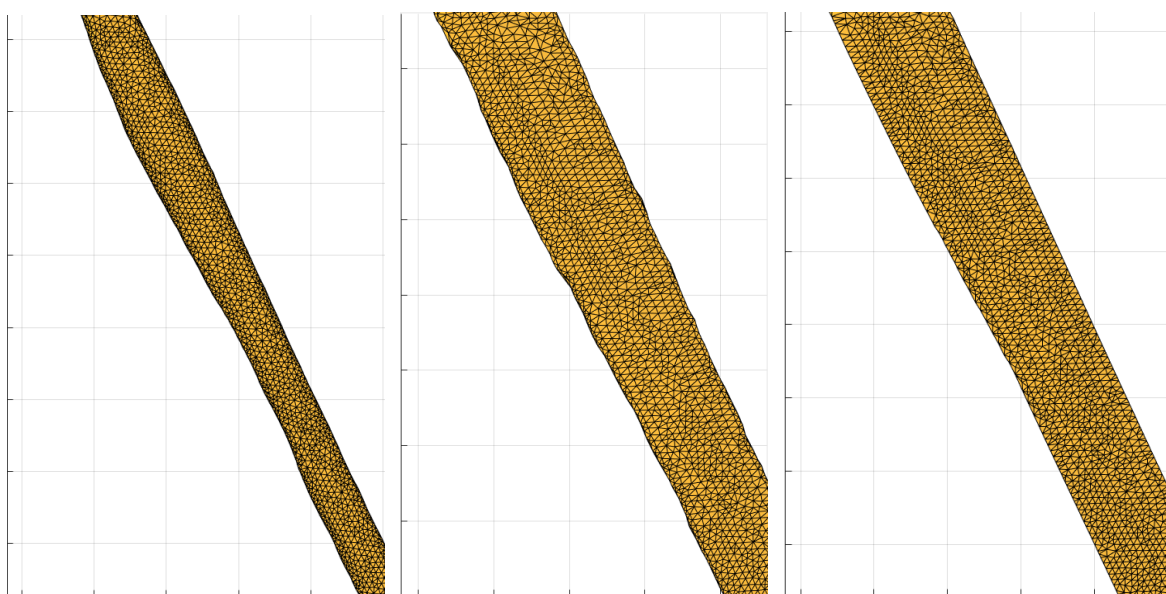
Proces Laplacoveho odšumovania má za úlohu znížiť mieru šumu v polygonálnej sieti. Na rozdiel od predchádzajúcej metódy, ktorá je v základnej podobe jednokroková je táto metóda vždy opakovaná po určitý počet iterácií N . Algoritmus odšumovania posúva každý bod siete

v smere ťažiska jeho susedných bodov, vid' kapitola 1.3.2. Magnitúdu tohto posunutia určuje parameter λ .

Ako bolo spomenuté, Laplacovo odšumovanie v základnej podobe, nakoľko posúva body vždy v smere ťažiska okolitých bodov (a nie v smere normál), má tendenciu zmršťovať sieť, čo je zreteľné hlavne na okrajových hranách siete – vid' obr. 39.

Aplikáciou Taubinovho odšumovania sa zmršťovanie minimalizovalo (vid' obr. 40), avšak nebolo odstránené úplne. Namiesto zmršťovania však Taubinovo odšumovanie zvýraznilo nízko-frekvenčný šum – okrajové hrany trate obsahujú zvlnenia. Táto vlastnosť je spomenutá aj v [8].

V tomto prípade je však možné využiť charakter nasnímaných dát. Nakoľko jediné ostré hrany prítomné v nasnímaných dátach sú priamo okrajové hrany trate (ktoré sú zo sieťového vyjadrenia ľahko dostupné), je možné cieľ Taubinovho odšumovania zadefinovať aj manuálne. Aby sa okrajové hrany nezmršťovali a nevnikli, je možné im zmenšiť, resp. úplne obmedziť posun v rovine XY a aplikovať základné Laplacovo odšumovanie. Týmto sa vyhne pobočným efektom Taubinovho odšumovania (vid' obr. 41).

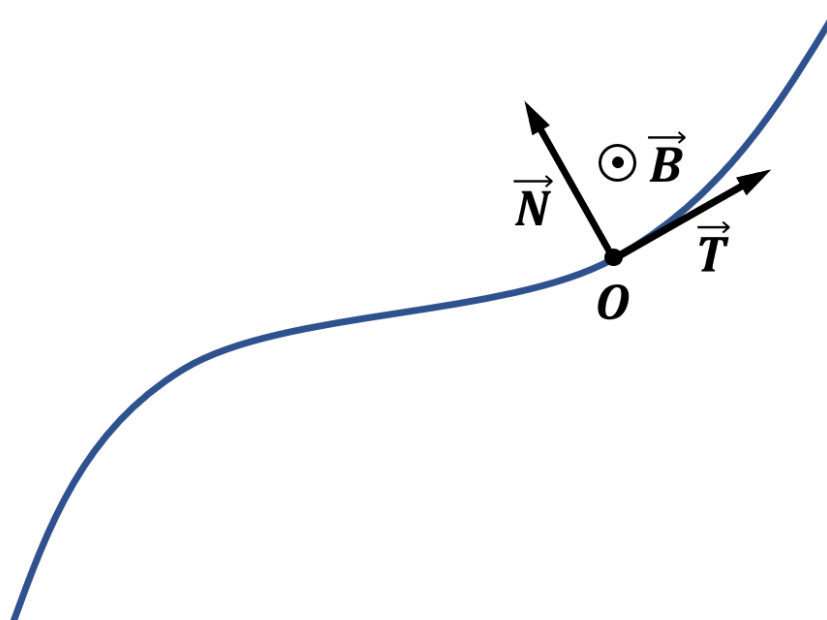


Obr. 39 Zmršťovanie siete Laplacovým odšumovaním Obr. 40 Taubinovo odšumovanie Obr. 41 Obmedzenie posunu hrán s Laplacovým odšumovaním

3.4 PRÍSTUP 3 – REKONŠTRUKCIA POMOCOU PRIRODZENÉHO SÚRADNÉHO SYSTÉMU

Doteraz sa pracovalo so štandardným kartézskym súradným systémom. Alternatívnym prístupom je kompletná rekonštrukcia trate za pomoci prirodzeného súradnicového systému vid' obr. 42. Tento prístup sa opiera o skutočnosť, že skenovaným objektom je trať, tj. nejaká definovaná trajektória. Pre takýto typ dát je taký prístup vhodný. Ponúknutý algoritmus prvotne vytvorí prirodzený súradný systém trate, do ktorého potom zaradí všetky body mračna. Počiatok sa riadi aktuálnym ťažiskom všetkých štyroch kolies, tangenciálna os aktuálnym smerovaním trate, normálová os aktuálnym sklonom vozovky okolo tangenciálnej osi.

Binormálová os je os kolmá na tieto dve osi a vyjadruje výšku, smer kolmý na povrch trate. Každý zo súradných systémov predstavuje určitú oblasť. Profil trate v každom z týchto miest je následne rekonštruovaný preložením polynómu cez nasnímané body.



Obr. 42 Prirodzený súradný systém všeobecnej priestorovej krivky. O – počiatok, T – tangenciála, N – normála, B – binormála

3.4.1 ZOSTAVENIE SÚRADNÉHO SYSTÉMU

Prvým krokom daného algoritmu je zostavenie súradného systému. Súradný systém bude vyjadrený štyrmi maticami:

$$Origins = [n \times 3], \quad (17)$$

$$Tangents = [n \times 3], \quad (18)$$

$$Normals = [n \times 3], \quad (19)$$

$$Binormals = [n \times 3], \quad (20)$$

Prvá matica vyjadruje súradnice počiatkov prirodzených súradných systémov vyjadrených v kartézskych súradniciach, ďalšie tri matice vyjadrujú jednotkové vektory jednotlivých osí prirodzeného súradného systému (tangenciála, normála, binormála), vyjadrené v kartézskych súradniciach.

Súradný systém by mal pokryť všetky miesta trate ale pritom by sa nemalo stať, že budú kolidovať medzi sebou. Pokiaľ je trať prechádzaná takým spôsobom, že počas jedného celého prejazdu sa neskenuje žiadne miesto viac ako raz, druhá podmienka nenastane. Pokiaľ sa však trať skenuje dôkladnejšie a niektoré miesta sú prejdené viac ako raz počas jedného celého prejazdu (čo je aj prípad daného vstupného súboru dát), je nutné regulovať vytváranie súradného systému.

Ako základ pre tvorenie súradného systému je zvolený prvý plný prejazd trate – tj prvé dosiahnutie miesta, z ktorého skenovanie začínalo (tzn. že proces skenovania by mal začínať na trati, inak je nutné manuálne začiatkové body mimo trate odstrániť – v opačnom prípade by algoritmus nedetekoval začiatok a vyšetroval by všetky body všetkých prejazdov).

Samotný algoritmus tvorenia súradného systému postupuje takto:

1. Algoritmus postupne prechádza všetky nasnímané body od začiatku a pre každý nasnímaný okamžik zapíše počiatok súradného systému a smery všetkých troch súradných osí prirodzeného súradného systému. Počiatkom súradného systému bude vždy ťažisko všetkých 4 kolies – stred auta. V každom z týchto bodov je rovnako počítaná vzdialenosť k prvému zapísanému počiatku (začiatok trate). Pokiaľ je táto vzdialenosť menšia ako určený limit, vozidlo prešlo jeden plný okruh a vytváranie súradného systému končí.
2. V ďalšom kroku je súradný systém prefiltrovaný pomocou parametru „res“ ako je to ukázané na obr. 43. Týmto parametrom sa určí požadované rozlíšenie trate v jej tangenciálnom smere. Filtrovanie súradného systému taktiež zabezpečí, aby každej oblasti prislúchal dostatočný počet bodov pre následné preloženie polynómom, čím sa zabezpečí konzistentnosť algoritmu. Taktiež sú prefiltrované počiatky na miestach, ktoré prechádza vozidlo viackrát.

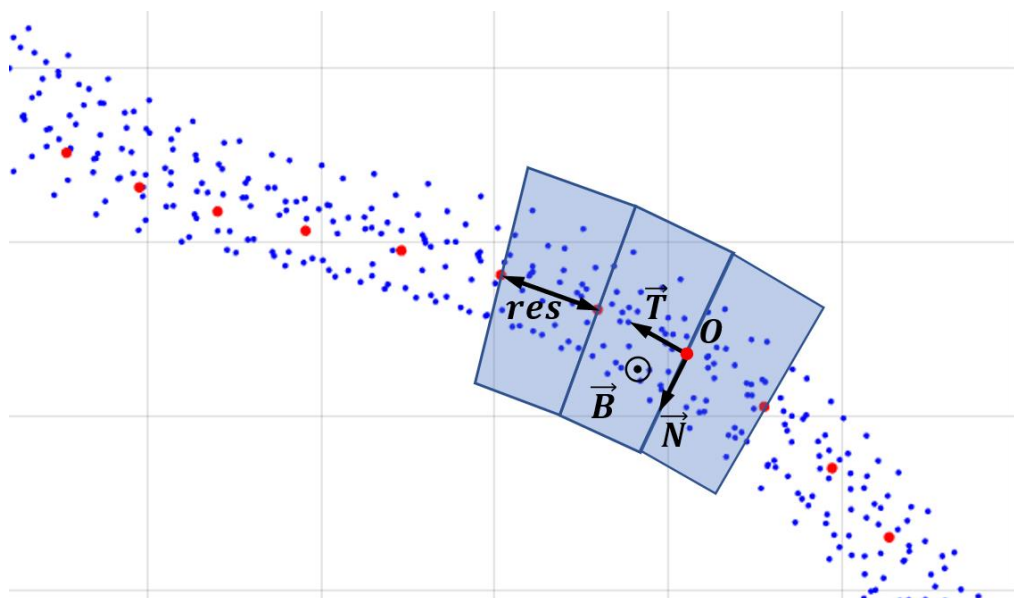
Takto vznikne systém súradných systémov pozdĺž celej dĺžky trate, pričom každý bod bude môcť byť jednoznačne priradený iba jednému súradnému systému (oblasti).

3.4.2 PRIRADENIE BODOV K JEDNOTLIVÝM OBLASTIAM

Zostavením prefiltrovaného súradného systému vznikne súbor oblastí, ktoré patria vždy jednému súradnému systému. Podobne ako u metódy filtrovania pomocou voxelovej siete aj tu nasleduje priradenie bodov k týmto oblastiam.

Pre konzistentnosť a efektivitu metódy sú priradované body iba vyjadrujúce stopu ľavého predného kolesa a ľavého zadného kolesa. Body pravej stopy sú potom priradené k rovnakým oblastiam ako body ľavého predného kolesa a ľavého zadného kolesa. Ak by tomu tak nebolo, mohlo by dochádzať k falošnému skloňovaniu trate vplyvom priradenia ľavého a pravého kolesa iným oblastiam.

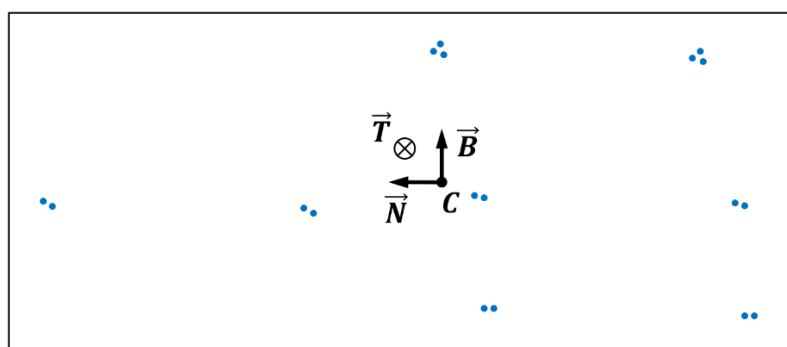
Prvý bod sa priradí prehľadom všetkých boxov. Po tom ako je zaradený, nasledujú ostatné body. Najprv algoritmus skontroluje, či je bod v úseku, v ktorom bol predchádzajúci bod. Pokiaľ nie, kontroluje nasledujúci úsek. Pokiaľ nepatrí ani tam, je kontrolovaný predchádzajúci úsek – v niektorých miestach je pohyb vozidla opačný smeru indexovania súradného systému. Pokiaľ bod nepatrí ani tam, sú prehľadované všetky úseky.



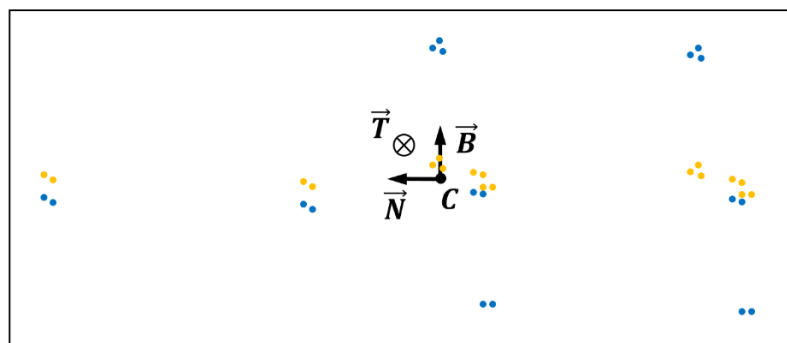
Obr. 43 Prirodzený SS na danom mračne bodov – červené body sú počiatky súradných systémov. Modré štvoruholníky vymedzujú oblasti do ktorých sú zaraďované body.

3.4.3 REKONŠTRUKCIA TRATE

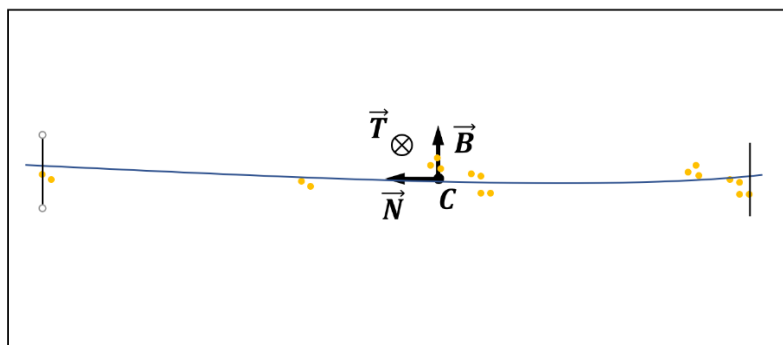
Body sú následne v každom úseku preložené polynómom druhého stupňa (v rovine kolmej na tangenciálny smer), ktorý vyjadruje aproximovaný tvar vozovky a na základe ktorého sa priamo určia body trate. Ešte predtým je však potrebné redukovať šum v binormálovom smere. To sa vykoná posunutím bodov o stanovenú hodnotu k priemernej hladine výšky v danej oblasti.



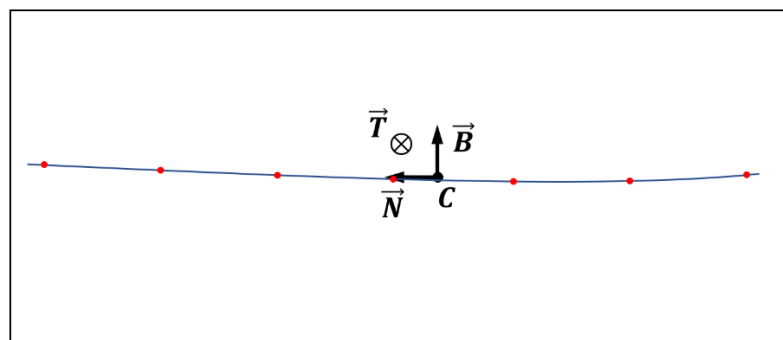
Obr. 44 Body priradené oblasti – pohľad v tangenciálnom smere



Obr. 45 Korekcia polohy bodov v binormálovom smere



Obr. 46 Preloženie bodov polynómom 2. stupňa a určenie okrajov trate



Obr. 47 Rekonštrukcia trate pomocou zistenia funkčných hodnôt polynómu

3.4.4 TRIANGULÁCIA DÁT

Triangulácia súboru bodov prebieha rovnako ako v kapitole 3.2.5.

3.5 POROVNANIE A VYHODNOTENIE JEDNOTLIVÝCH METÓD REKONŠTRUKCIE

Predstavené metódy sú od seba výrazne odlišné a je vhodné ich vzájomne porovnať a zhrnúť ich výhody a nevýhody.

3.5.1 ROBUSTNOSŤ

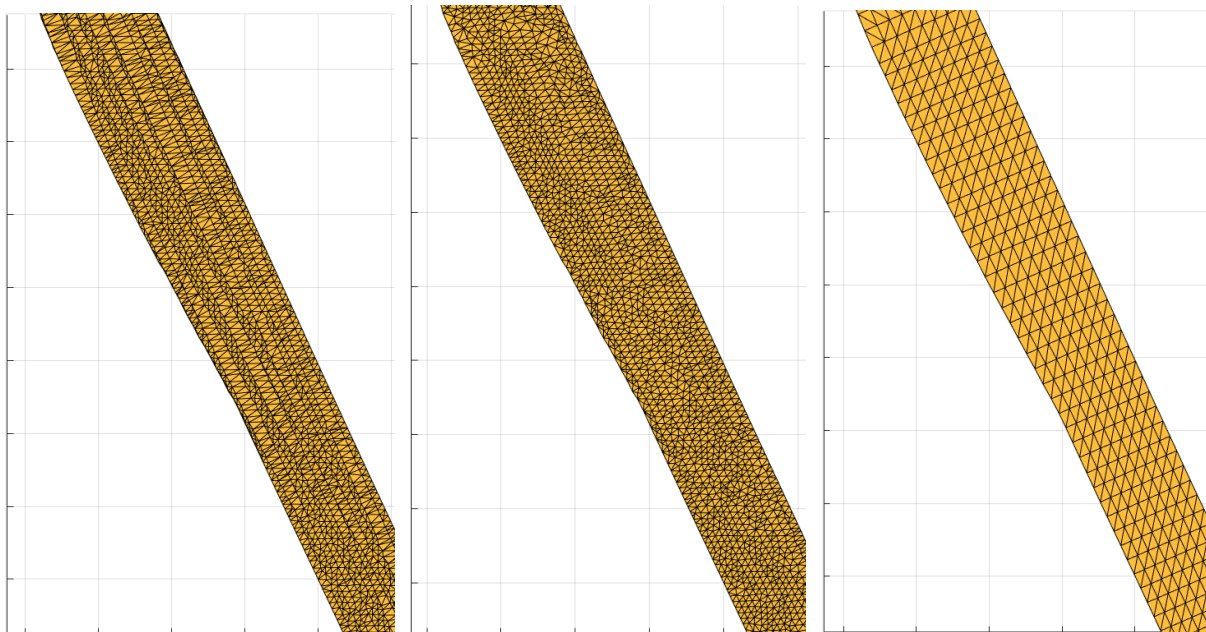
Kritériom, ktoré vyjadruje vhodnosť a univerzálnosť použitej metódy je kritérium robustnosti. Vysokou univerzálnosťou sa vyznačujú metódy 1 a 2, ktoré je možné aplikovať prakticky na akýkoľvek typ dát, nemusí ísť čisto o vyjadrenie trate. Na druhej strane metóda 3 je vhodná iba na traťový typ dát, nakoľko algoritmus je postavený na použití prirodzeného súradného systému ktorý je vhodný pre definované trajektórie.

3.5.2 DOSIAHNUTÉ ODŠUMENIE

Z hľadiska odšumovania je zložité posudzovať metódy nakoľko nie sú rovnocenné a každá z nich je nastaviteľná vzhľadom na žiadané odšumenie. Napriek tomu je možné predpokladať, že najmenej šumu bude obsahovať sieť rekonštruovaná podľa metódy 3, nakoľko výšky bodov sú funkčnými hodnotami polynómov 2. stupňa, tj. body nasledujú jedno plynulé zakrivenie. Taktiež v prípade prístupu 2 je dosiahnuté odšumenie lepšie regulovateľné ako v prípade prístupu 1, nakoľko prístup 2 je iteratívny a okrem odšumovacích parametrov je možné aj zvýšiť alebo znížiť počet iterácií pre dosiahnutie požadovaného stupňa odšumenia.

3.5.3 ROVNOMERNOSŤ SIETE

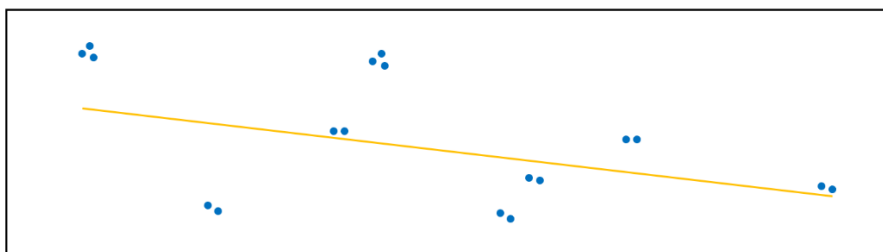
Ďalším kvalitatívnym hľadiskom pri posudzovaní siete môže byť aj rovnomernosť siete, tj. variabilita veľkosti prvkov siete. Vzhľadom na princíp rekonštrukcie, tretia metóda disponuje najrovnomernejšou sieťou nakoľko body sú rozmiestnené samotným algoritmom – jedná sa o kompletnú rekonštrukciu. Nakoľko v prvej metóde bilaterálneho filtrovanie sa body pri odšumovaní posúvajú výhradne v smere normál, druhý prístup – Laplacovo odšumovanie je z tohto hľadiska lepší, nakoľko body sa posúvajú v smere ťažísk ich susedných bodov – nakoľko ťažisko je geometrickým stredom, algoritmus má prirodzenú tendenciu zrovnomerňovať sieť.



Obr. 48 Rovnomernosť siete po odšumení. Zľava – prístup 1. V strede – prístup 2. Sprava – prístup 3.

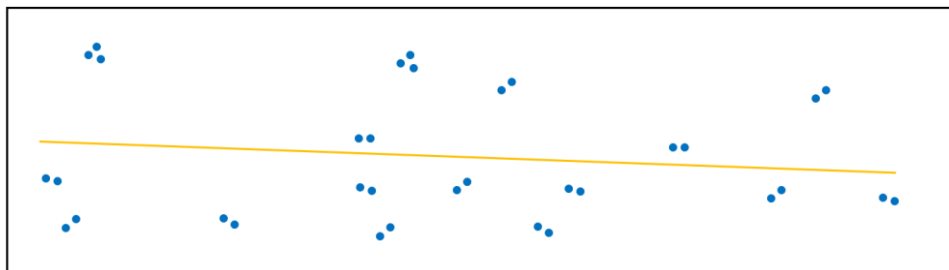
3.5.4 VPLYV ŠUMU

Najviditeľnejším pobočným efektom šumu je prítomný sklon vozovky okolo tangenciálnej osi – v miestach kde sa skombinovalo Z – odchýlenie s odlišnou pozíciou prejazdu v rovine XY, trať má tendenciu sa sklápať, viď obr. 49. Tento efekt je najviac prítomný v prípade prístupu 1.



Obr. 49 Sklápanie trate – pohľad v tangenciálnom smere. Žltým označený približný smer výslednej plochy

Spôsobom akým sa tento efekt dá potenciálne zmierniť je použitie väčšieho dátového súboru s viacerými prejazdami rovnakého miesta.



Obr. 50 Sklápanie trate – väčší dátový súbor. Žltým označený približný smer výslednej plochy

4 NÁVRH DOPLNENIA OKOLITÉHO TERÉNU

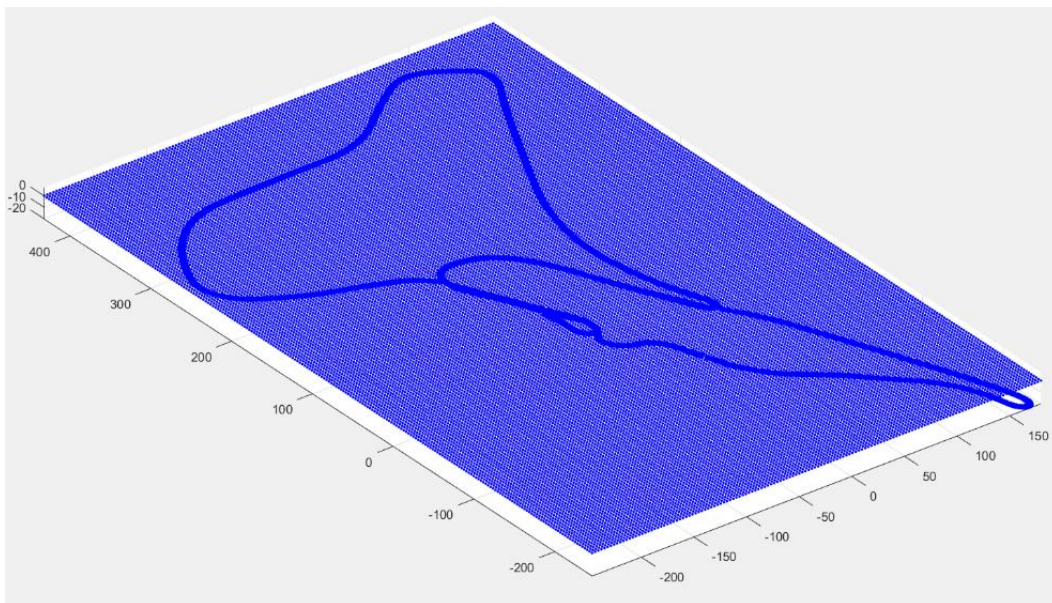
Pre lepšiu vizualizáciu terénu je vhodné doplniť do modelu trate aj okolie trate, ktoré však nie je súčasťou súboru zosnímaných dát – daná skenovacia metóda sníma iba miesta, ktoré vozidlo fyzicky prešlo. To môže byť problém pri miestach okolo trate, ktoré však zvyčajne nevyžadujú takú presnosť ako vyjadrenie vozovky. Preto je možné ich doplniť aj alternatívnymi metódami. Budú predstavené dva odlišné návrhy: doplnenie pomocou výškových máp a vizualizačné doplnenie. Výber záleží na tom, aký cieľ sa týmto doplnením sleduje.

4.1 VIZUALIZAČNÉ DOPLNENIE

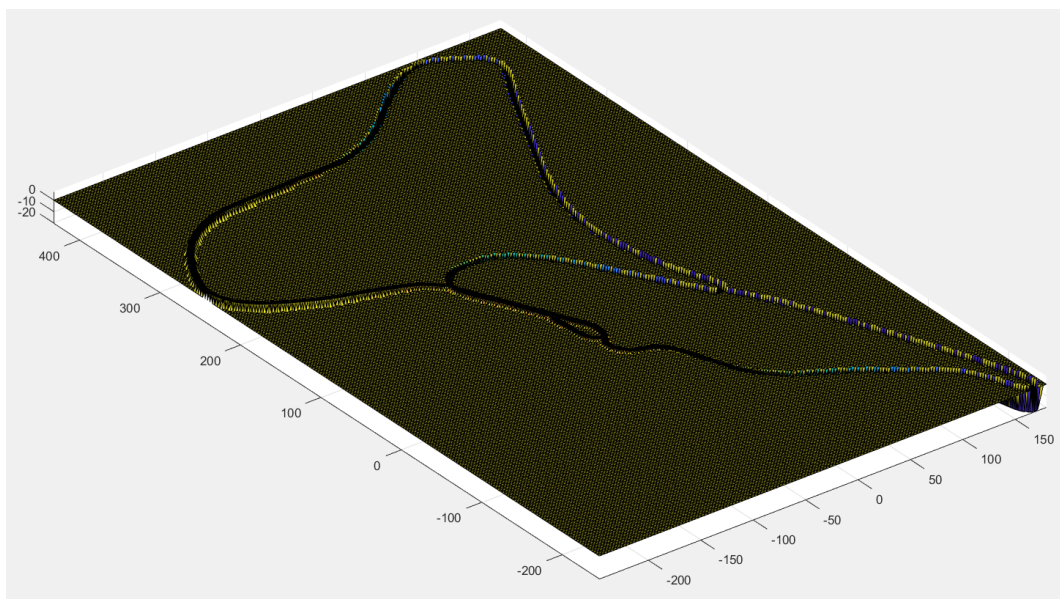
Pokiaľ je cieľom doplnenie okolitých miest za účelom lepšej vizualizácie trate a nie je nutné cieľom zachytiť okolitý reliéf presne podľa skutočnosti, najjednoduchším spôsobom doplnenia je doplniť nezosnímané miesta všeobecným reliéfom okolia.

Metóda využitá v tomto prípade vychádza z iteratívneho Laplacoveho odšumovania. Vytvorí sa množina okolia trate, ktorá je sprvu tvorená sieťou bodov v jednej zvolenej výškovej rovine, vid' obr. 51. Následne sa táto množina bodov spojí s množinou bodov trate, po čom je postup veľmi podobný klasickému Laplacovému odšumovaniu, s niekoľkými odlišnosťami:

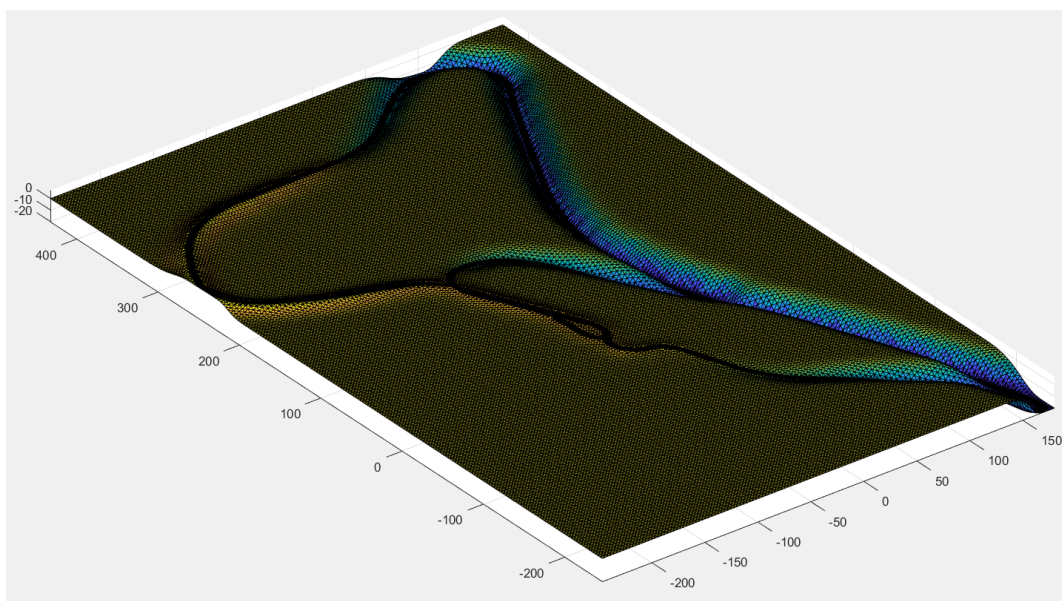
- odšumuje sa iba tá časť siete, ktorá vyjadruje okolie trate – samotná trať sa neodšumuje, ale je súčasťou siete a je uvažovaná pri nachádzaní susedných vrcholov prvého rádu
- nevyužije sa Taubinova metóda ale štandardná Laplacova metóda s jedným parametrom λ , ktorý však v tomto prípade nie je užívateľom definovaný (nakol'ko je pre efektivitu nastavený na maximálnu hodnotu)
- odšumovanie prebieha výhradne v Z-smere aby sa predišlo modifikácii hustoty siete a jej celkového tvaru



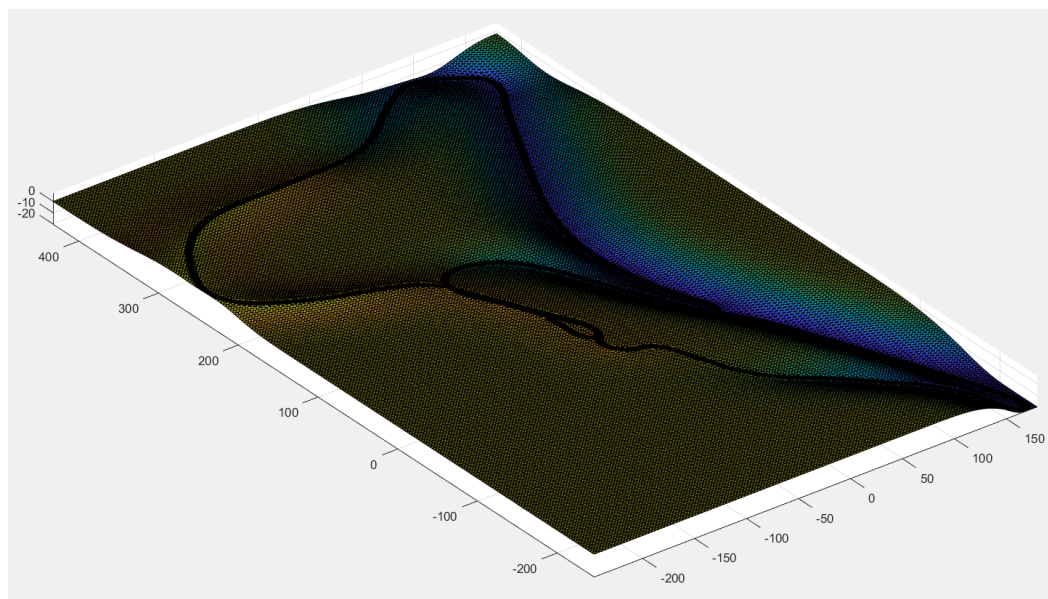
Obr. 51 Dva súbory bodov – trať a okolie



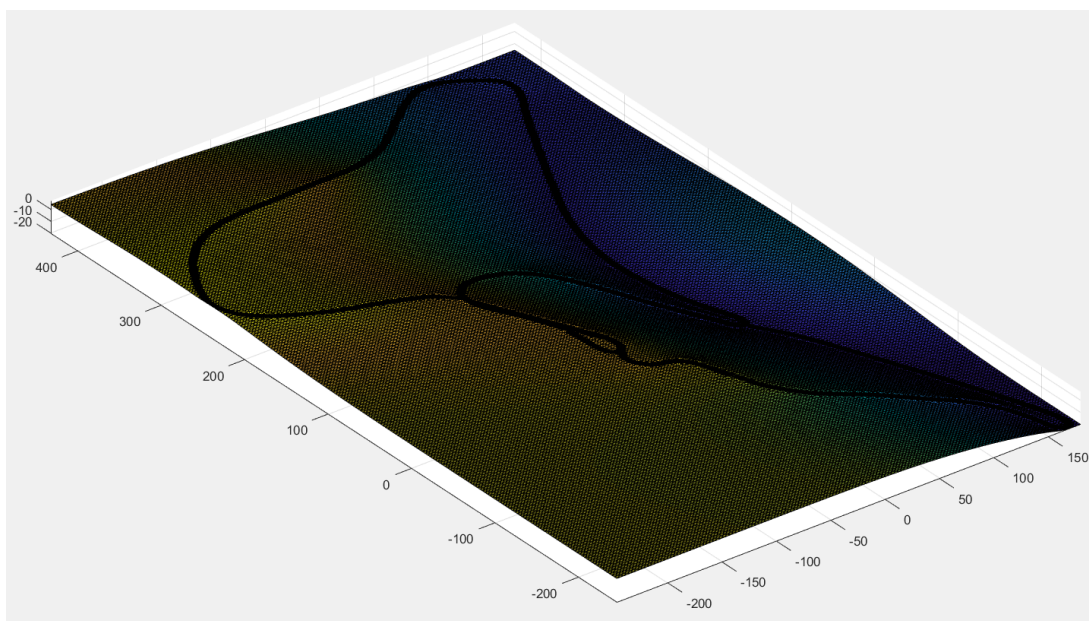
Obr. 52 Triangulácia spoločnej množiny trate a okolia



Obr. 53 Trat' a okolie po $N = 20$ odšumovacích iteráciách



Obr. 54 Trať a okolie po $N = 200$ odšumovacích iteráciách



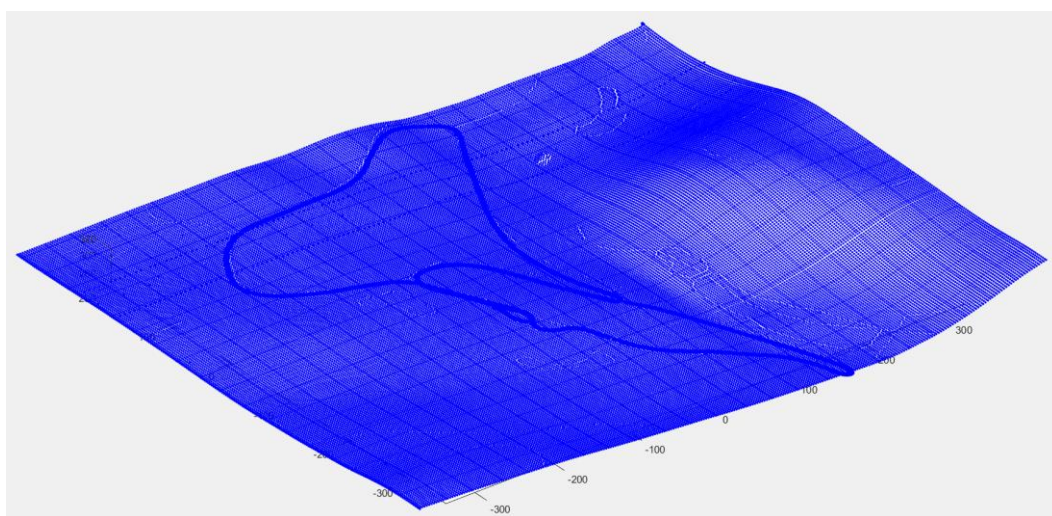
Obr. 55 Trať a okolie po $N = 2000$ odšumovacích iteráciách

Výhodou tohto prístupu je rýchle doplnenie s minimálnym vstupom od užívateľa, ktorý spočíva prakticky iba v zadaní požadovaných parametrov. Týmto parametrami sú výška roviny obsahujúcej body okolia, hustota týchto bodov a počet iterácií algoritmu – vplyv počtu iterácií na tvar doplneného terénu je viditeľný na obr. 53 – obr. 55.

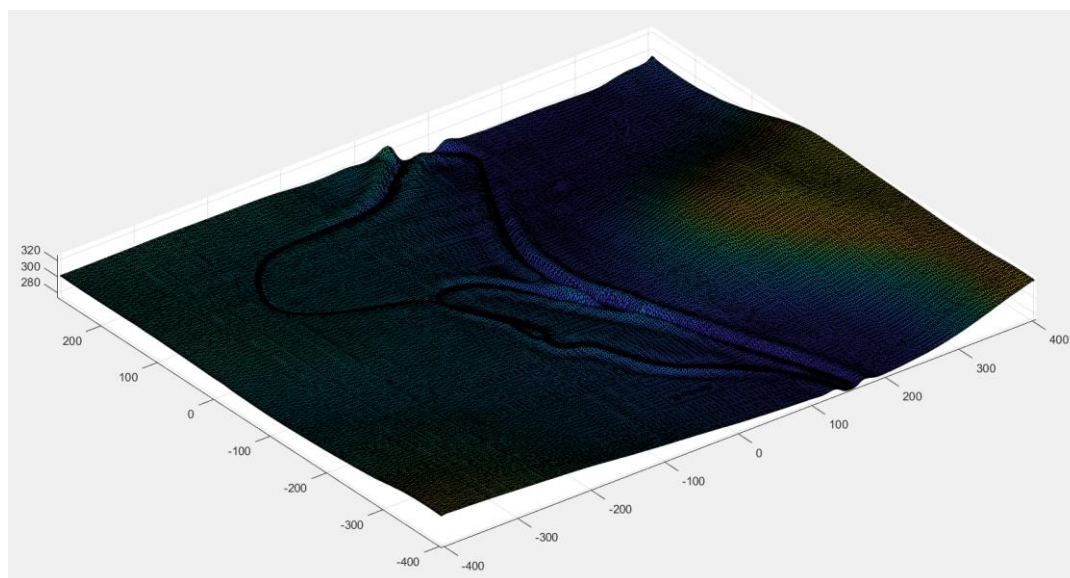
4.2 DOPLNENIE POMOCOU VÝŠKOVÝCH MÁP

Pokiaľ je však cieľom získať aj do určitej miery realistický reliéf okolia, je možné siahnuť po voľne dostupných výškových mapách zemského povrchu a rekonštruovať okolie trate práve pomocou týchto máp.

Tu je však možné sa stretnúť s problémom, nakoľko tieto voľne dostupné mapy terénu nemajú dostatočné plošné rozlíšenie pre spoľahlivé doplnenie terénu. Pokiaľ by bol rozdiel malý, stačilo by spojiť súbor bodov trate a body extrahované z výškovej mapy dokopy a triangulovať. V prípade trate skúmanej v tejto práci však majú výškové mapy veľkú výškovú deviáciu oproti nasnímanej mape – v niektorých miestach viac ako 10 m, čo sa žiaľ zväčša nedá vhodne nastaviť posunutím alebo škálovaním. Situácia sa však môže líšiť prípad od prípadu, v tomto prípade je však nutné pre spojenie týchto dvoch množín bodov pristúpiť na metódu ich plynulého prepojenia. Metóda použitá v tomto prípade bude veľmi podobná ako metóda použitá v predchádzajúcej kapitole 4.1., avšak s tým rozdielom, že v tomto prípade bude použité Taubinovo odšumovanie, nakoľko je snaha zachovať výškový profil okolia trate bez jeho zmršťovania.



Obr. 56 Dva súbory bodov – trať a okolie



Obr. 57 Trať a okolie po $N = 200$ odšumovacích iteráciách

Tento prístup však z princípu vyžaduje manuálny vstup, nakoľko je potrebné získať výškovú mapu a plošne ju posunúť do správnej pozície a prípadne aj naškálovať vzhľadom na nasnímanú trať.

5 POUŽITIE NÁSTROJA

Obr. 58 Užívateľské prostredie

Vytvorené užívateľské prostredie umožňuje jednoduchý prístup ku všetkým dostupným funkciám. Užívateľ môže v rámci tohto prostredia načítať súbor bodov, vybrať si metódu odšumovania, spustiť odšumovanie so zvolenými parametrami a vykresliť výslednú triangulovanú plochu. Túto plochu je možné po zadefinovaní názvu ihneď uložiť vo formáte .stl, prípadne ju ešte doplniť vizualizačným doplnením o okolitý terén. Taktiež je možné vizualizovať vstupný neodšumený súbor bodov. V prípade zmeny parametrov spojených so samotným odšumovaním sa dá „prepočítať“ výsledok, čím sa vynechajú prípravné kroky algoritmu a spustí sa iba samotný proces odšumovania.

Postup pri práci s užívateľským prostredím je nasledovný:

1. Užívateľ vyberie súbor bodov pomocou tlačidla „Vybrať súbor“. Súbor môže byť z akéhokoľvek priečinka ale musí byť vo formáte .mat, .xls alebo .xlsx a musí sa jednať o maticu s veľkosťou $n \times 12$.
2. Tlačidlom „Zobraziť súbor bodov“ môže užívateľ vykresliť načítaný súbor bodov.
3. V zozname „Odšumovacia metóda“ si užívateľ môže vybrať z troch metód popísaných v kapitole 3. a prípadne vybrať aj metódu hľadania najbližších bodov (platí iba pre prístup 1 – bilaterálne filtrovanie). Taktiež môže nastaviť dostupné parametre. Parametre sa zobrazujú a miznú podľa aktuálne vybranej metódy.

4. Keď sú parametre nastavené, môže užívateľ spustiť proces odšumovania tlačidlo „Spustiť“. Proces potrvá nejaký čas a zobrazí sa dialógové okno s aktuálnym stavom procesu. Akonáhle proces skončí, v novom okne sa automaticky vykreslí výsledná plocha po odšumovaní.
5. Pokiaľ je užívateľ spokojný, môže priamo exportovať túto plochu do formátu .stl pomocou tlačidla „Vytvoriť STL“. Musí však prv zadať názov v políčku nad týmto tlačidlom.
6. Pokiaľ nie je spokojný s výsledkom, môže zmeniť akékoľvek parametre a spustiť proces znovu. Pokiaľ budú zmenené parametre, ktoré sú priamo spojené s odšumovacím procesom, samotný odšumovací proces môže byť spustený pomocou tlačidla „Prepočítať“. Tým sa ušetrí čas nakoľko sa nespustia znova prípravné algoritmy pred odšumovaním ale iba samotné odšumovanie. Pri zmene ostatných parametrov alebo pri zmene odšumovacej metódy / spôsobu hľadania najbližších bodov sa musí proces spustiť znova pomocou tlačidla „Spustiť“.
7. Pokiaľ si užívateľ želá doplniť aj okolie trate, môže tak učiniť pomocou tlačidla „Doplniť okolie“. Môže nastaviť výšku a hustotu okolia a počet iteračných krokov.
8. Pokiaľ je potreba nastaviť trať do správnej nadmorskej výšky (napr. kvôli doplneniu pomocou výškových máp), slúži na to sekcia „Nastavenie trate“. Tlačidlom „Ukázať extrém“ sa v novom okne otvorí súbor bodov a zobrazí sa na ňom výškové minimum a maximum. Pomocou ľubovoľnej online mapovej služby je potrebné zistiť nadmorské výšky v týchto dvoch miestach a vpísať ich do príslušných koloniek MAX a MIN. Po zadaní týchto hodnôt stačí už iba stlačiť tlačidlo „Nastaviť výšku“, čím sa súbor bodov nastaví a naškáluje na príslušné nadmorské výšky.

ZÁVER

Rekonštrukcia trate so sebou prináša rôzne výzvy, riešenia ktorých boli predstavené v tejto práci. Pomocou vypracovaného nástroja je možné nasledovať celý postup rekonštrukcie od odšumovania bodov a vytvorenia polygonálnej siete, cez doplnenie okolia až po vytvorenie výstupného STL súboru. Každá trať je však iná a existuje množstvo faktorov ktoré ovplyvňujú úspešnosť použitých metód redukcie šumu a rekonštrukcie ako takej, taktiež niektoré metódy sú vhodné pre niektoré situácie pre ktoré sú druhé metódy menej vhodné. Preto je užívateľovi umožnené zadanie najdôležitejších parametrov ktoré sprevádzajú tieto procesy pre individuálne prispôsobenie podľa aktuálnej situácie.

POUŽITÉ INFORMAČNÉ ZDROJE

- [1] REIMPELL, Jornsens. *The Automotive Chassis*. 2nd edition. Oxford: Butterworth - Heinemann, 2001. 444 s. ISBN 0-7506-5054-0
- [2] GILLESPIE, Thomas. D. *Fundamentals of Vehicle Dynamics*. Warrendale: Society of Automotive Engineers, 1992. 519 s. ISBN 1-56091-199-9.
- [3] VARSHOSAZ, Masood, Davood SHOJAEI a H. HELALI. *The Methods of Triangulation* [online]. 2005 [cit. 2019-05-23]. Dostupné z: https://www.researchgate.net/publication/273458201_The_Methods_of_Triangulation
- [4] EBRAHIM, Mostafa Abdel-Bary. *3D Laser Scanners' Techniques Overview*. International Journal of Science and Research (IJSR) [online]. 2015, , 323-331 [cit. 2019-05-23]. ISSN 2319-7064. Dostupné z: https://www.researchgate.net/publication/282753883_3D_Laser_Scanners'_Techniques_Overview
- [5] MA, Zhiliang a Shilong LIU. *A review of 3D reconstruction techniques in civil engineering and their applications*. Advanced Engineering Informatics [online]. 2018, (37), 163-174 [cit. 2019-05-23]. Dostupné z: https://www.researchgate.net/publication/326745151_A_review_of_3D_reconstruction_techniques_in_civil_engineering_and_their_applications
- [6] TOMASI, C. a R. MANDUCHI. *Bilateral Filtering for Gray and Color Images* [online]. 1998 [cit. 2019-05-23]. Dostupné z: https://www.researchgate.net/publication/3766290_Bilateral_Filtering_for_Gray_and_Color_Images
- [7] DIGNE, Julia a Carlo DE FRANCHIS. *The Bilateral Filter for Point Clouds*. Image Processing On Line [online]. 2017, (7), 278-287 [cit. 2019-05-23]. ISSN 2105-1232. Dostupné z: https://www.researchgate.net/publication/320703396_The_Bilateral_Filter_for_Point_Clouds
- [8] OHTAKE, Yutaka, Alexander BELYAEV a Ilia BOGAEVSKI. *Mesh regularization and adaptive smoothing*. Computer-Aided Design [online]. 2001, (33), 789-800 [cit. 2019-05-23]. Dostupné z: https://www.researchgate.net/publication/223004724_Mesh_regularization_and_adaptive_smoothing
- [9] TAUBIN, Gabriel. *Curve and Surface Smoothing Without Shrinkage*. IEEE International Conference on Computer Vision [online]. 1995, , 852-857 [cit. 2019-05-23]. Dostupné z: https://www.researchgate.net/publication/3612986_Curve_and_Surface_Smoothing_Without_Shrinkage
- [10] EDELSBRUNNER, Herbert a Ernst MUCKE. *Three-Dimensional Alpha Shapes*. ACM Transactions on Graphics [online]. 1994, (13), 43-72 [cit. 2019-05-23]. Dostupné z: https://www.researchgate.net/publication/2137834_Three-Dimensional_Alpha_Shapes

- [11] PORTEŠ, P.; ZHÁŇAL, L.; FOJTÁŠEK, J. *Complex approach to computations and analysis of vehicle dynamics*. In *Vibroengineering Procedia*. Vibroengineering Procedia. Lithuania: JVE International, 2018. p. 132-137. ISSN: 2345-0533.
- [12] LOUIS BENTLEY, Jon. *Multidimensional Binary Search Trees Used for Associative Searching*. *Communications of the ACM* [online]. 1975, (18), 509-517 [cit. 2019-05-23]. Dostupné z: https://www.researchgate.net/publication/220424200_Multidimensional_Binary_Search_Trees_Used_for_Associative_Searching_Communications_of_the_ACM_189_509-517
- [13] *Thin out a huge point cloud with the pointcloud class* [online]. [cit. 2019-05-23]. Dostupné z: <https://www.geo.tuwien.ac.at/downloads/pg/pctools/publish/pointCloudThinOut/html/pointCloudThinOut.html>
- [14] *Bunny Wireframe* [online]. [cit. 2019-05-23]. Dostupné z: <http://www.cs.mun.ca/~omeruvia/philosophy/WireframeBunny.html>
- [15] *Delauney Triangulations and the Traveling Salesman* [online]. [cit. 2019-05-23]. Dostupné z: <https://web.colby.edu/thegeometricviewpoint/2015/03/09/delauney-triangulations-and-the-traveling-salesman/>
- [16] *Point Set Triangulations* [online]. [cit. 2019-05-23]. Dostupné z: <https://commons.wikimedia.org/wiki/File:PointSetTriangulations.svg>
- [17] *Interpreting map features* [online]. [cit. 2019-05-23]. Dostupné z: <http://www.bushwalking101.org/interpreting-map-features/>
- [18] *Heightmap test* [online]. [cit. 2019-05-23]. Dostupné z: <https://gamedev.amazon.com/forums/storage/attachments/5275-heightmap-test.png>
- [19] *Cubic Structure and Floor Depth Map* [online]. [cit. 2019-05-23]. Dostupné z: <https://www.flickr.com/photos/dominicspics/5393871227/in/photostream/>
- [20] *Strom (informatika)* [online]. [cit. 2019-05-23]. Dostupné z: [https://commons.wikimedia.org/wiki/File:Strom\(informatika\).jpg](https://commons.wikimedia.org/wiki/File:Strom(informatika).jpg)
- [21] *Podstrom* [online]. [cit. 2019-05-23]. Dostupné z: <https://commons.wikimedia.org/wiki/File:Podstrom.jpg>
- [22] *Princíp merania* [online]. [cit. 2019-05-23]. Dostupné z: <http://vlm.sjf.stuba.sk/podstranky/3D%20principy.html>
- [23] *Low Cost Solid-State LiDAR* [online]. [cit. 2019-05-23]. Dostupné z: <https://hackaday.io/project/162579-low-cost-solid-state-lidar>
- [24] BONETTO, Sabrina, Anna FACELLO, Anna Maria FERRERO a Gessica UMILI. *A tool for semi-automatic linear feature detection based on DTM*. *Computers & Geosciences* [online]. 2014, (75) [cit. 2019-05-23]. Dostupné z: https://www.researchgate.net/publication/268751511_A_tool_for_semi-automatic_linear_feature_detection_based_on_DTM

- [25] SAHILLIOĞLU, Yusuf. [25] *Mesh smoothing Observation: shrinkage problem* [online]. [cit. 2019-05-24]. Dostupné z: <https://slideplayer.com/slide/9266522/27/images/10/Mesh+smoothing+Observation%3A+shrinkage+problem..jpg>
- [26] *Creating a DataPointsFilter* [online]. [cit. 2019-05-23]. Dostupné z: <https://libpointmatcher.readthedocs.io/en/latest/DataPointsFilterDev/>

Zoznam Použitých Skratiek a Symbolov

c	$[\text{m}\cdot\text{s}^{-1}]$	Rýchlosť svetla
d	$[\text{m}]$	Vzdialenosť od snímača k snímanému bodu
d_d	$[\text{m}]$	Euklidovská (geometrická) vzdialenosť bodu Q k bodu P
d_n	$[\text{m}]$	Vzdialenosť bodu Q k regresnej ploche v bode P
d_x, d_y, d_z	$[\text{m}]$	Veľkosti voxelov na každej z osí súradného systému
k	$[-]$	Stanovený počet najbližších susedných bodov
N	$[-]$	Počet iterácií odšumovacieho procesu
n_x, n_y, n_z	$[-]$	Počty voxelov na každej z osí súradného systému
P	$[\text{m m m}]$	Odšumovaný bod
P_d	$[\text{m m m}]$	Odšumený bod
Q	$[\text{m m m}]$	Susedný bod bodu P
r	$[\text{m}]$	Stanovený polomer hľadania najbližších susedných bodov
t	$[\text{s}]$	Čas od vyslania lúča po jeho návrat
λ	$[-]$	Odšumovací parameter pre Laplacovské metódy odšumovania
μ	$[-]$	Odšumovací parameter pre Taubinovo a Bilaplaciálne odšumovanie
σ_d	$[-]$	Gaussova váha pre euklidovskú (geometrickú) vzdialenosť
σ_n	$[-]$	Gaussova váha pre vzdialenosť k regresnej ploche

ZOZNAM PRÍLOH

Prílohou k tejto práci je komprimovaný súbor Rekonstrukcia.zip, ktorý obsahuje tieto súbory:

1. Rekonstrukcia.mlapp – hlavný spúšťací súbor, ktorý ovláda všetky ostatné funkcie
2. affine_fit.m
3. bilateralDenoise.m
4. bilateralPrepare.m
5. buildVoxelGrid.m
6. dataPreprocessing.m
7. draw.m
8. fill1.m
9. getmaxmin.m
10. intrinsic.m
11. kd_buildtree.m
12. laplacianDenoise.m
13. laplacianPrepare.m
14. readFile.m
15. setHeight.m
16. triangulate.m
17. VoxelGridFiltering.m
18. VoxelNeighbours.m